Inteligencia Artificial, 20(59) (2017), 82-95
doi: 10.4114/intartif.vol20iss59pp82-95

INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

An Orientation Method with Prediction and Anticipation Features

João Ramos[1], Tiago Oliveira[1], Ken Satoh[2], José Neves[1], Paulo Novais[1]
[1]Algoritmi Centre, Department of Informatics
University of Minho
Braga, Portugal
{jramos,toliveira,jneves,pjon}@di.uminho.pt
[2]National Institute of Informatics
Hitotsubashi, Chyoda-ku
Tokyo, Japan
ksatoh@nii.ac.jp

Abstract Nowadays, progress is constant and inherent to a living society. This may occur in different arenas, namely in mathematical evaluation and healthcare. Assistive technologies are a topic under this evolution, being extremely important in helping users with diminished capabilities (physical, sensory, intellectual). These technologies assist people in tasks that were difficult or impossible to execute. A common diminished task is orientation, which is crucial for the user autonomy. The adaptation to such technologies should require the minimum effort possible in order to enable the person to use devices that convey assistive functionalities. There are several solutions that help a human being to travel between two different locations, however their authors are essentially concerned with the guidance method, giving special attention to the user interface. The CogHelper system aims to overcome these systems by applying a framework of Speculative Computation, which adds a prediction feature for the next user movement giving an anticipation ability to the system. Thus, an alert is triggered before the user turn towards an incorrect path. The travelling path is also adjusted to the user preferences through a trajectory mining module.

Keywords: Orientation system, Speculative computation, Trajectory data mining, Localization system.

1 Introduction

Cognitive disability is a broad concept which includes different intellectual or cognitive deficits. These deficits may be present from birth (like birth defect) or may be acquired later (like traumatic brain injury). More precisely, this term is used to define a person who has more difficulties in one or more types of mental tasks when compared to an ordinary person [19]. A disability may be present in several levels of incidence, varying from mild to extreme. An individual with severe or extreme cognitive disabilities needs constant assistance throughout his everyday life whereas a mild to moderate disabled person may be capable of having an independent life, only requiring some assistance in certain activities, which may be provided by a caregiver or a technological system.

Assistive technology aims to increase, maintain or even improve functional capabilities of a person with disabilities [1]. A mental task commonly affected is orientation, which is imperative for an independent life. Thus, it is necessary to have technologies that assist the user during his travel between home and

ISSN: 1137-3601 (print), 1988-3064 (on-line)

office/school. Using an orientation device the user is sufficiently autonomous to travel between his current location to a predefined destination. Current approaches focus essentially on the guidance activity, giving more attention to the information display and to the communication with a caregiver [5, 12, 9]. In order to be capable to correctly use such applications there is the need of a training period in which the person with cognitive disabilities has to lear how to use the application. It has been proven that, despite the type of cognitive loss, a person may learn how to use different types of technology [8].

This work proposes an orientation system for the early stages of cognitive disabilities (mild and moderate) that, besides guiding the user, tries to anticipate possible mistakes. An alert is triggered when the user is expected to make a wrong turn in his path. In order to adapt the system to the user it is also included a trajectory mining feature so it is possible to calculate a path that is preferred by the user (which may not be the shortest one).

This paper is organized as follows. Section 2 presents related work concerning orientation systems for people with cognitive disabilities. Section 3 provides a description of the orientation system giving emphasis to new developments on the Speculative Module and Trajectory Mining Module. The Speculative Module that hosts the framework for Speculative Computation is explained in Section 4. On Section 5 the Trajectory Module used to get a path according to the user's preferences is described. Finally, conclusions are drawn and future work considerations are made in Section 6.

2 Related Work

The technological development of smartphones brought more portability to the user since it became possible to execute applications in small and portable devices. These devices are specially important to people with cognitive disabilities since it is through them that the user may contact his caregiver. In order to execute an application to guide the user, the developers have to pay special attention to the interface [8]. This must be simple to understand and to interact with, otherwise the developed application may have a low acceptance degree due to the necessary cognitive effort to use it.

The works described in [12], [5] and [9] are examples of three different orientation methods for people with cognitive disabilities. In these examples the main goal is to guide a person outdoors from the current location to a predefined destination. The difference between them resides in how this is done, but all are particularly focused on the user interface. Thus, they lack the predictive capabilities which would allow them to anticipate wrong user actions and apply necessary measures to avoid them, and the capability of adjusting the path to user preferences. Despite the importance of such applications to people with cognitive disabilities, this research area has received little attention from the academic and industrial community. Thus, as far as we know, there is not any recent work considering this investigation topic.

With the goal of guiding cognitive disabled people and considering the systems interface, Carmien *et al.* [5] developed an application that enables the user to travel using a public transportation system reducing the effort needed to understand complex transportation maps (Figure 1). On each bus a GPS unit was installed and its coordinates were transmitted to a remote server. Using the information from the server the system would be able to select the bus that the user should take in order to reach the intended destination. While the user is travelling, a personal travel assistant ensures that the user has taken the correct bus, alerting him otherwise.

Liu *et al.* [12] focused their research on the display of instructions (Figure 2). They used static pictures with overlaid arrows (or highlighted areas in an image), audio and/or visual messages to guide the user (Figure 2a). Their objective was finding the best way of providing the directions to the user using either static pictures reflecting user perspectives with landmarks that are easy to find or visual/audio messages whenever an image is not available (Figure 2b). According to the necessary travelling distance the authors discovered that a distant landmark should be more useful since it could be used for a longer time while closer landmarks needed to be replaced more often according to the user movements.

A different approach was used by Fraunhofer Portugal in the AlzNav orientation system [9]. This system used an arrow that resembles a compass to guide the user (Figure 3). This arrow rotates according to the direction in which the mobile device is pointed. Thus, the user has to interpret the information presented on the device's screen. Besides the compass, the user has also information about the street he is and the distance he should travel in the calculated direction. This system has also a monitoring system enabling caregivers to know the current position of the user through a SMS. A "safe area" may also be

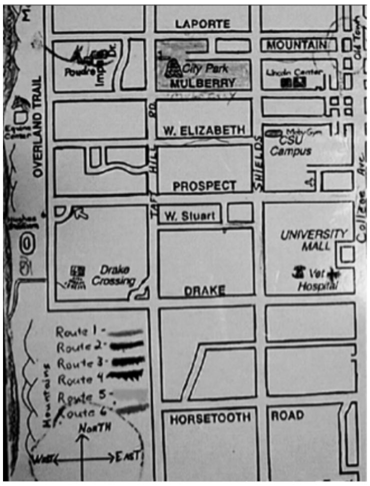


Figure 1: Complex map of the routes of the public transportation system [5]

set by the caregiver in where the user may travel alone and an alert is only triggered when he goes out of it.

The previously presented approaches tackle some important aspects of orientation systems for people with cognitive disabilities. However there are some features that should be considered in order to make the system adaptable to the user and not the other way around. Predicting user steps is a big advantage for this type of system since it is possible to identify critical points in a certain path, i.e., identify points where the user takes the wrong path and alert the user before he makes a mistake. If the system is able to predict when an error will occur, it can issue an alert to the user reinforcing the right path. Another important feature is the ability to adjust the path to the user since he may prefer to travel for a longer but preferred path instead of taking the shortest one. These are the kind of features proposed in this work. The goal is to develop an orientation system that adapts to the user, maximizing his autonomy and consequently his independence.

3 System Description

CogHelper is an ongoing project [17, 15, 16] with two main goals: provide an efficient orientation system for people with cognitive disabilities, considered the main user of the system, and provide a tracking system for caregivers. The former is accomplished through an augmented reality interface so users only need to align the mobile device with the correct travelling path in order to see a green arrow indicating

(a) Different types of landmarks

(b) Different ways to guide the user

Figure 2: Information display using landmarks [12]

Figure 3: Interface for the alignment of the AlzNav application [9].

the right direction. Here, both the prompting level and the travelling path is adjusted according to the user preferences. The latter enables caregivers to know at any time the current position of the person with disabilities.

This orientation system is devised for outdoor and the system architecture is depicted in Fig. 4, which may be separated in the client-side applications and the server-side components. The former includes the *Cognitive Helper Mobile Solution* (described in detail in Section 3.1), the *Caregiver Applications (Mobile and Web)* (described in Section 3.2), and the *External Services*, which allow the integration of CogHelper with other systems or applications, adding more features/functionalities to it. The latter is divided into the database module, which stores all the information necessary for the correct execution of the system (like usernames, locations and points of interest), the web services, and the remaining module enables the communication between the server and the different services.

The orientation method under development is conceived for outdoors which has been fully described in [14]. The core of the system is considered to be fully developed, *i.e.*, both applications for caregivers (mobile and web), the web services (running on the server) needed in order to ensure communications between applications, the database, and finally the mobile application for the person with cognitive disabilities. At this point of the development this application uses augmented reality for the orientation and the selected path is the shortest one (not being adapted to the user as proposed in this paper).

3.1 Application for People with disabilities

The primary target of the CogHelper system is people with cognitive disabilities. Thus, the mobile application intended to this audience is composed by four layers (see Fig. 5), each with specific functions. The *Information Layer* stores the information for the normal execution of the application (like user data and his contacts). In the *Localization Layer* the current user position is retrieved from the GPS module of the device (or from the network) and is used by the *Navigation Algorithm*. To this information is added data gathered from the device sensors, like the camera, magnetic sensor and accelerometer (enabling the system to compute the device's direction), which are used by the decision algorithm to ensure the user is

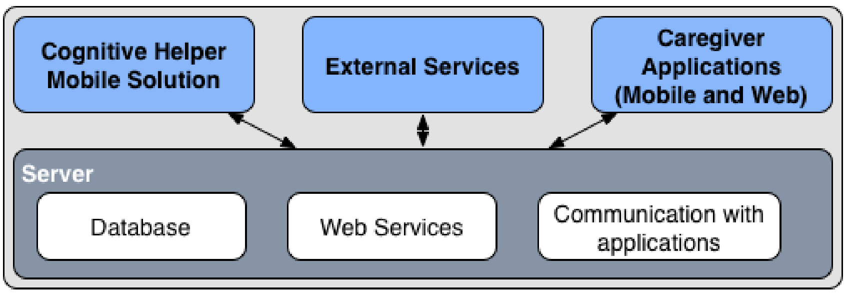


Figure 4: Architecture of CogHelper System

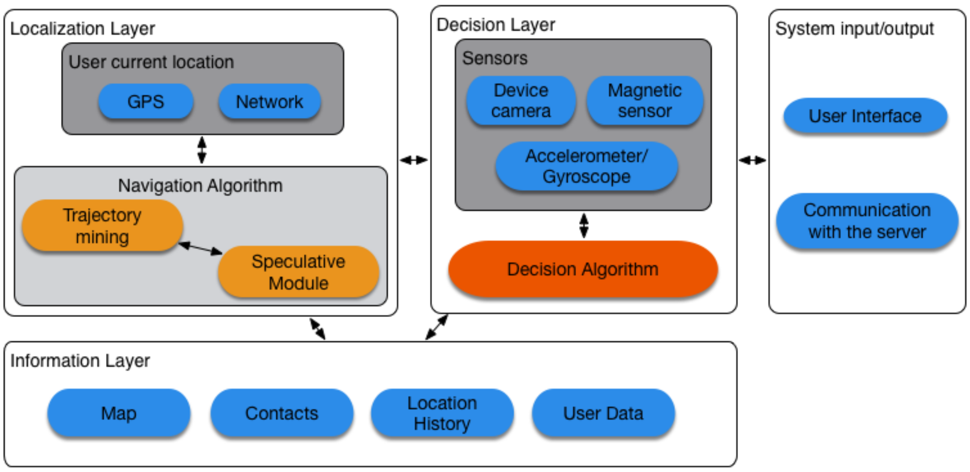


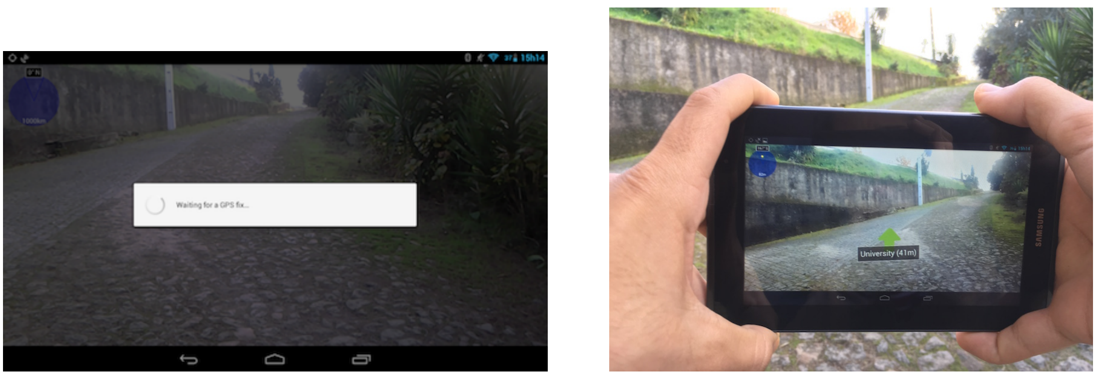
Figure 5: Information layers of the CogHelper Module for People with Cognitive Disabilities

travelling in the correct path (Figure 6). All information is then presented to the user through the user interface (under the *System input/output*). For an easier interpretation of the displayed information, the guiding process has an augmented reality interface so the user just has to orientate the device with the correct travelling path in order to view a green arrow pointing the correct path. When the user goes out of the path (even if he does it intentionally) the systems alerts the caregiver that the user may be lost.

A detailed description of previously cited modules is done in [14, 15]. Being an ongoing project, CogHelper is being improved with new modules in order to give the system an adaptability feature. The *Trajectory mining* and *Speculative Module* components (depicted in Fig. 5) are responsible for the adaptability of the system to the user.

The *Trajectory mining* component generates a path that is preferred by the user (which may not be the shortest one), *i.e.*, the path is calculated according to the preferences of the user (historic data from previous uses of the application). This path is used as input values of the *Speculative Module*, which ensures that the user is travelling in the correct path alerting him otherwise. These modules are described in more detail in Section 5 and Section 4, respectively.

Moreover, one cannot exclude the development of additional features, such as the ones described in [6], in order to detect other user activities, namely fall detection.



(a) Waiting for a GPS signal (b) Informing the path to the user

Figure 6: Orientation method for people with disabilities - Augmented reality interface

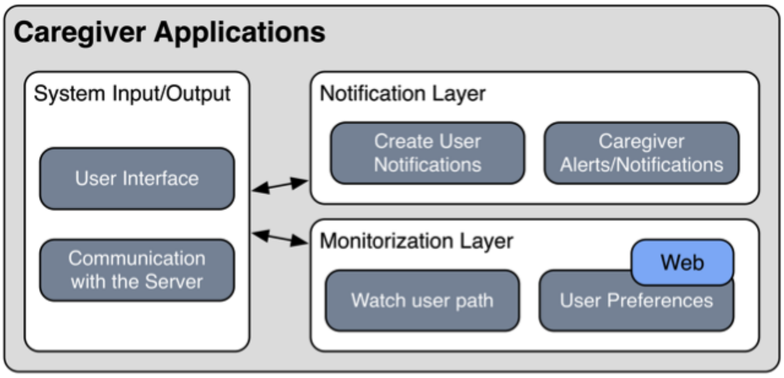


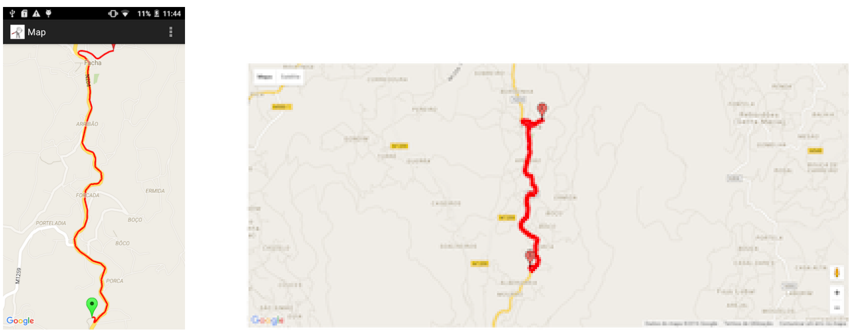
Figure 7: Information layers of the CogHelper Module for Caregivers

3.2 Applications for Caregivers

Caregivers may not be the primary target of CogHelper system but they should not be dismissed. If a caregiver may not know the user status, he may become worried and prevent the user to travel alone and autonomously between two locations. In order to let caregivers be an active part of the orientation system, CogHelper also supports a localization system enabling caregivers to know in real time the current location of the user. This feature enables caregivers to develop a different activity without neglecting the care provided to the user. Indeed, both caregivers and users have their autonomy increased.

There are two different applications intended to caregivers, a mobile application for Android OS and a Web application. Developed for two different platforms, these applications have the same goal: provide a localization system. Indeed, these two applications provide a monitoring platform in which caregivers may remotely check the current position of the user. There is no need for the caregiver to be physically with the user. Despite the shared goal, there are some differences between the two. As shown in Fig. 7 the dissimilarity resides under the *Monitorization Layer* in which the *User Preferences* module is only present in the Web application. Through this module the caregiver may add, edit or remove user preferences such as his destination points or the available options (giving or denying access, for example, to the option for manual insertion of the destination). These additional features make the Web application more complex.

Despite the previously described difference, the architecture of the applications for the caregivers is divided in three layers: *Monitorization Layer*, *Notification Layer* and the *System Input/Output*. The *Watch User Path* module allows the caregiver to remotely have access to the current location of the user. The *Notification Layer* is composed by two modules which receive and generate alerts from and to the user application. The remaining layer is made by the *User Interface*, that enables the caregiver to interact with the applications, and a communication module to exchange information with the server.



(a) Android application (b) Web application

Figure 8: Localization system for caregivers

On Fig. 8 is depicted an example of a path viewed in the mobile and web applications, respectively. On both examples the starting and ending points are marked with different symbols, enabling an easier identification of these locations. When the user is travelling the ending mark does not appear and the map view is updated to show his current location.

4 Speculative Module for Users with Cognitive Disabilities

The Speculative Module under the mobile application for people with cognitive disabilities has the objective of predicting the next step of the user (when using the orientation system) and use that prediction to set the information that should be displayed to the user (alert/warning or acknowledge messages). The execution of the Speculative Computation resembles an interface between the rules with the instructions for the correct path, the set of default values (predictions about user travels from one location to another, obtained from the trajectory mining module - Section 5), and real information returned by information sources (informing the real journey of the user).

Through the use of this module the system continues its execution using a default value (whenever the real information is missing) or using the real one (returned from the information sources). Thus, the system does not enter an idle state when there is missing information. It tries to generate a tentative solution for the problem, which is revised when the real information is received (to verify if the default value is consistent with the real one). The default values are obtained before the execution of the speculative computation framework from the Trajectory Pattern Mining described in Section 5.

For the execution of the Speculative Computation module the computation changes between its normal execution phase (*Process Reduction Phase*) and temporarily to a revision phase (*Fact Arrival Phase*) to revise the computation according to the received values. The initial information, before execution, represented in the Speculative Computation framework includes:

1. All the possible paths between two points, in the form of connections between intermediary points, as facts in the knowledge base;

2. The transitions between points usually performed by the user as default values;

3. Information of whether a point is included or not in the recommended path as default values;

4. A set of rules that structure the derivation of the path the user is likely to follow given the information during execution and the issuing of alerts/warnings in case of a potential mistake;

At the beginning of the computation, when there is no information regarding the actual position of the user and his transitions between the most relevant points, the defaults are used in the *Process Reduction Phase* to build the most likely path, step by step, and issue the warnings for potential mistakes or acknowledgements of correctly taken steps. A warning is issued whenever a user is likely to take the wrong path, which may happen when the defaults tell the computation that the user will make a transition to a point not included in the correct path. Through *Fact Arrival*, the GPS sensor and a recognizer inform the Speculative Module of the actual transitions of the user and whether the points are indeed part of the correct path or not. If the user actually moves to a point not included in the correct path, the recognizer re-calculates the path to the destination and a point previously out of the correct path may suddenly become part of the new path. *Fact Arrival Phase* is the mechanism through which this information is updated and the tentative paths produced for a user are adjusted and improved.

Items 1, 2, and 3 from the list above are obtained from a Trajectory Mining Module. Item 1 corresponds to the calculation of the possible paths between two points, producing a reduced graph, with only the most relevant points an the connections between them. Item 2 is obtained from the pattern mining of the trajectories usually taken by the user, reflecting his walking habits. Finally, item 3 corresponds to the calculation of the recommended path between the point of origin and the destination, expressed in the form of intermediary points included in the route. It is also stated which points are not included in this route.

The Framework of Speculative Computation in the Orientation Method for people with cognitive disabilities (designed by SF_{OM}) is defined in terms of the signature $\langle \Sigma, \mathcal{E}, \Delta, \mathcal{A}, \mathcal{P}, \mathcal{I} \rangle$ [18], where:

- Σ stands for a finite set of constants (an element of Σ is called a system module);
- $\mathcal{E}$ denotes a set of functions called *external predicates*. When Q is a literal belonging to an external predicate and S is the identifier of the information source, $Q@S$ is called an *askable literal*. We define $\sim$*(Q@S)* as *(*$\sim$*Q)@S*;
- Δ designates the *default answer* set, which is a set of ground askable literals that satisfy the condition: Δ does not contain both $p(t_1,\ldots,t_n)@S$ and $\sim\ p(t_1,\ldots,t_n)@S$ at once;
- $\mathcal{A}$ is a mark of a set of predicates called *abducible predicates*. Q is called *abducible* when it is a literal with an *abducible predicate*;
- $\mathcal{P}$ signals a set of rules of the form:
  - ▷ $H \leftarrow B_1, B_2, \ldots, B_n$ where H is a positive ordinary literal, where each of $B_1,\ldots,B_n$ is an ordinary literal, an askable literal or an abducible; and
  - ▷ H is the head of rule R and is named as *head(R)* (always non-empty), being R the rule of the form $H \leftarrow B_1,\ldots,B_n$; $B_1,\ldots,B_n$ is the body denoted by *body(R)*, that in some situations is substituted by the boolean value *true*.
- $\mathcal{I}$ is a set of integrity constraints of the form:
  - ▷ $\bot\leftarrow B_1, B_2, \ldots, B_n$, where $\bot$ is a contradiction special symbol and B_1, B_2, $\ldots$, B_n is an ordinary literal or an *askable literal* or an *abducible*. However at least one of $B_1, B_2, \ldots, B_n$ is an *askable literal* or an *abducible*.

An *askable literal* may have different meanings, namely:

1. An *askable literal* $Q@S$ in a rule $\mathcal{P}$ stands for a question put to a system module S by OM; and
2. An *askable literal* in Δ denotes a default truth value, either *true* or *false*, i.e., $p(t_1,\ldots,t_n)@S \in \Delta$, $p(t_1,\ldots,t_n)@S$ is usually *true* for a question to a system module *S*, and $\sim p(t_1,\ldots,t_n)@S \in \Delta$, $p(t_1,\ldots,t_n)@S$ is generally *false* for a question to a system module *S*.

As an example, one can consider the graph of Figure 9, in which the objective is for the user to move from node 1 to node 3. From the Trajectory Mining Module it is possible to know that the user usually moves from 1 to 2, from 2 to 3, but also from 3 to 4, from 4 to 5, from 5 to 6, and from 6 to 3. Additionally, it is possible to determine that nodes 1, 2, and 3 are part of the shortest path, while node 4 is obviously not. However, since the user, when in node 2, usually moves to 4, this is identified during *Process Reduction* as a point where a mistake may happen and, as a result, a warning is issued. If, during *Fact Arrival*, it is confirmed the user indeed moves to 4, this node becomes now part of the route and the alternative path including 5 and 6 is selected. Although simple, this example illustrates the role of the Speculative Module in producing instructions for the user and preventing mistakes in his path.

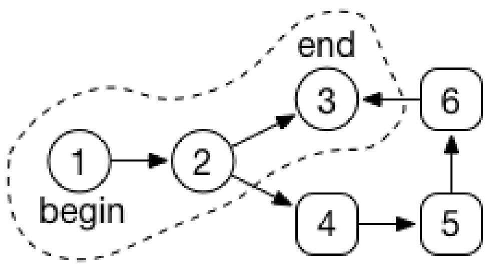


Figure 9: Possible ways to travel between locations 1 and 3.

Below there is a logic program that represents the formalization of the problem depicted in Figure 9 and the situation described above according to the Speculative Computation Framework for Users with Cognitive Disabilities. Its components include: Σ, a representation of existing system modules responsible for providing information; $\mathcal{E}$, the predicates which represent the necessary information to derive the path of the user; Δ is the default answer set and consists in a set of default values; $\mathcal{A}$, a set of abducible predicates; and $\mathcal{P}$ is a logic program with a set of clauses. In the logic program given below the literal $path(a, b)$ denotes that there is a physical connection between the locations a and b, thus the user may travel between them. The literal *show_next_point* is used to indicate that the system must show the next location to which the user should travel. This location may be an intermediate point or the final destination. Whenever the user travels in the wrong direction the literal *show_user_warning* is activated indicating to the system that it must alert the user. In the set $\mathcal{E}$ there are the predicates $user\_travel(a, b)$ (which states that the user will travel from location a to location b) and $included(a)$ (to indicate if a location a is part of the route). The values for these predicates are asked from the information sources *gps_sensor* and *recognizer*, respectively. The former verifies if the user is travelling from point A to B. The latter checks if point B is included in the set of valid locations.

- $\Sigma = \{gps\_sensor, recognizer\}$
- $\mathcal{E} = \{user\_travel, included\}$
- $\Delta = \{user\_travel(1, 2)@gps\_sensor, user\_travel(2, 3)@gps\_sensor,$
  $user\_travel(2, 4)@gps\_sensor, user\_travel(4, 5)@gps\_sensor,$
  $user\_travel(5, 6)@gps\_sensor, user\_travel(6, 3)@gps\_sensor,$
  $included(1)@recognizer, included(2)@recognizer,$
  $included(3)@recognizer, \sim included(4)@recognizer$
  $included(5)@recognizer, included(6)@recognizer,$
- $\mathcal{A} = \{show\_next\_point, show\_user\_warning\}$
- $\mathcal{P}$ is the following set of rules:
  $guide(A, A) \leftarrow .$
  $guide(A, B) \leftarrow$
  $path(A, F),$
  $show\_next\_point(F),$
  $user\_travel(A, F)@gps\_sensor,$
  $guide(F, B).$
  $guide(A, B) \leftarrow$
  $path(A, F),$
  $user\_travel(A, F)@gps\_sensor,$
  $show\_user\_warning(F),$
  $guide(F, B).$
  $path(1, 2) \leftarrow .$
  $path(2, 3) \leftarrow .$
  $path(2, 4) \leftarrow .$
  $path(4, 5) \leftarrow .$
  $path(5, 6) \leftarrow .$
  $path(6, 3) \leftarrow .$
- $\mathcal{I}$ denotes the following set of integrity constraints or invariants:
  $\bot \leftarrow$
  $show\_next\_point(F),$
  $\sim included(F)@recognizer.$
  $\bot \leftarrow$
  $show\_user\_warning(F),$
  $included(F)@recognizer.$

For the Execution of the Speculative Module there are some important definition that must be done. During its execution the Framework is composed by a Process Set (PS), which represents a set of processes (active or suspended); a set of Already Asked Questions (AAQ) containing the askable literals; and the Current Belief State (CBS) is a set of askable literals. PS expresses all the alternative computations that were considered. The AAQ set is used to avoid asking redundant questions to the sensors. CBS is the current belief state and expresses the current status of the outside world.

A process defined as the tuple $\langle GS, OD, IA, ANS \rangle$ in which GS is a set of extended literals and called a Goal Set, expressing the current status of an alternative computation; OD is a set of askable literals called outside defaults, which denote a set of assumed information about the outside world during a computation; IA is a set of negative literals or abducibles called inside assumptions that stand for the values assumed during a computation; and ANS is a set of instantiations of variables in the initial inquiry.

At the beginning of the computation there is only an active process defined as $\langle \{guide(1,3), \emptyset, \emptyset, \emptyset\} \rangle$; AAQ is empty, since any question was asked to the sensors; and CBS is equal to the default values set. According to the definition described in [16] the computation enters in the *Process Reduction Phase*, in which it follows a set of rules in order to derive the next action. This is an iterative process where the set PS and AAQ are changed according to the computation state. After questioning the information source the computation uses default values and keeps its execution. When the information source returns the real value the CBS set is updated (if necessary) and the computation is revised, entering temporarily in the *Fact Arrival Phase*. In this phase, if the assumed default value is in accordance with the received one the computation continues, otherwise it is revised and processes using the default value enter in the suspend mode while suspended ones may be started. Then, the framework goes again to the *Process Reduction Phase* continuing the iterative process until there is no processes to reduce.

5 Trajectory Mining Module

Advances in mobile computation (*e.g.*, smartphones) and in location-acquisition methods (*e.g.*, GPS module) enabled the gathering of massive spatial trajectory data, which, in turn, has raised the interest of researchers in trajectory data mining [21]. According to [7] there are three important attributes when considering Behavioural Pattern Mining: location, trajectory and behaviour. The first attribute considers the extraction of important user locations (like home or office). The second one considers the trajectory modeling through the extraction of regular routes. The last attribute emphasizes the extraction of behavioural patterns. Thus the system may be able to predict the user's destination through his current path. By applying trajectory data mining we may get a preferable user path and use it as input of the previously described module of Speculative Computation.

Through active recording (the user location is logged only when the application is running) it is possible to obtain the position of the user. However there are a few steps that precede the trajectory pattern mining like trajectory data preprocessing and trajectory data management [21].

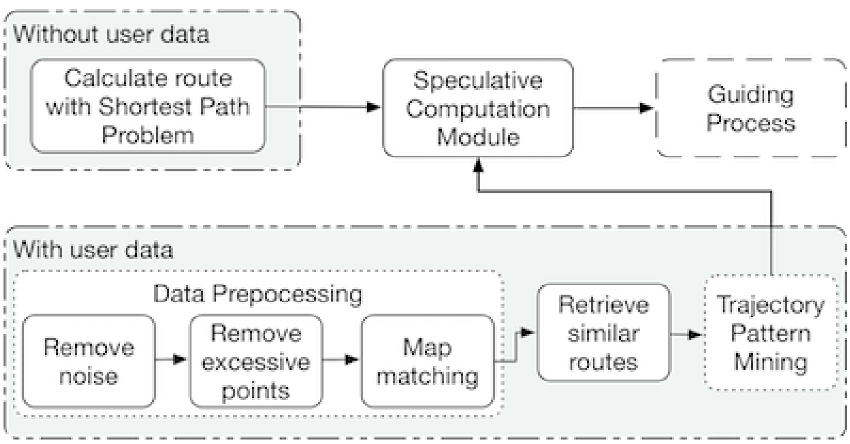
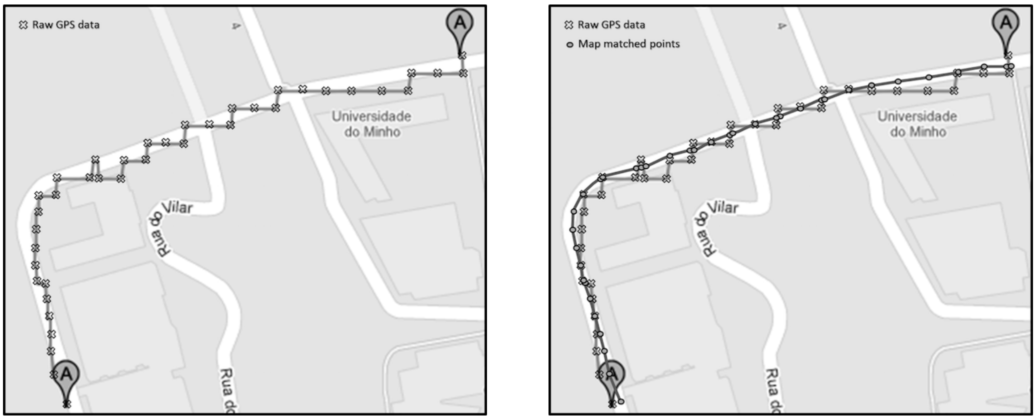


Figure 10: Generation of default values schema

For the mining module operation it is important to consider the existence of user historic data. When the user is using CogHelper for the first time (thus, there isn't any information about his travelling habits) the system calculates the shortest path and uses it as input for the speculative computation module in order to guide the user. When there is available information about the user, the mining module extracts similar routes according to the current user location and the intended destination and use them for the trajectory pattern mining. Ending this process the data is sent to the speculative computation module (as default values) so the system is able to guide the user. Before applying the pattern mining the retrieved data need to be preprocessed. An illustration of this process is presented in Figure 10.

5.1 Data Preprocessing

In a first step (trajectory data preprocessing) it is important to remove the noise from each collected position (Figure 11a is an example of raw data collected during a travel). There may be points that may appear outside the travelled route (due to a bigger GPS error), which should not be considered for the trajectory data mining (Figure 11b). To this step there are several on-line tools with map matching features (in our case we are using GraphHopper Map Matching API [11]). Then, since collected locations represent a large number of samples, it is important to remove excessive data, *i.e.*, points that do not bring useful information like intermediate points in roads without intersections.



(a) Route obtained from database raw data (b) Error removal from raw data

Figure 11: Trajectory data processing

After this process, the remaining data should represent locations on a map that may appear slightly outside the road. Thus, a map matching process is necessary in order to align the collected data with the existing real maps.

Ending this preprocessing stage and before starting the trajectory data mining process, the system has to obtain similar routes (*e.g.*, travel paths with the same destination and a similar starting point). Thus, instead of using all routes for this stage, the system selects the similar routes (or similar route parts) through similarity/distance functions. When, for a user, there is no route for an intended destination the system selects the shortest path.

5.2 Trajectory Mining

Finally, the trajectory data mining may begin. For this process there are different types of pattern mining. The described process may be considered as a standard for the use of trajectory data mining, however there might be some derivations of it. According to its own purposes an author may adapt this standard for his algorithm or system.

In [10] the authors consider a trajectory pattern as a set of individual trajectories which share an identical sequence of visited places. For their trajectory pattern mining in an initial stage the authors try to get a set of regions of interest (which is possible by different approaches) and then the authors try to define the trajectory patterns. Giannotti *et al.* intend to apply their trajectory pattern mining in the analysis of traffic flows. An extension of this work is presented in [13] in which the authors make use of all trajectories saved on the database to construct a predictive model in order to be able to predict the user's (or object's) next move (*e.g.*, predict where the user will be when the GPS module is temporarily unavailable).

A different approach is applied in [2]. Here the aim is to automatically obtain the frequent moves, neglecting the time at which they occur. To the authors a trajectory is an ordered list of stops and moves, *i.e.*, the user moves between two places (considered as stops) in which he stays for a given time interval. For their purpose the authors intend to discover the pattern of moves between two places usually done by the users regardless of the intermediate points (*e.g.*, the streets or roads used). Using this data Alvares *et al.* could obtain answers for questions like the most frequent stops during a period of time, which stops have a duration higher than a predefined threshold, among others.

Chen, in [7], proposed a model in which after the geo-coordinates extraction, a tree-based hierarchy graph is built. Through this the author intends to apply hierarchy density clustering algorithms like OPTICS [20] to find patterns in the recorded data. The OPTICS algorithm [20] has been used for mining people's life pattern using GPS log data. This is an algorithm that tries to find density-based clusters in the spatial data [4]. For this process a distance metric between location points is required in order to group the data into clusters.

Considering the different models and their application our goal is to define the best strategy to apply in CogHelper. According to our research and considering the goal for each method/algorithm, we consider that OPTICS will be the most appropriate for our system.

6 Conclusions and Future Work

This work defines how a trajectory mining method could be used to adapt the path to the user by producing a set of default values. These values are considered to be the predictions for the directions that the user should follow in order to travel between his current location to the intended destination. The trajectory mining module process uses data from previous executions of the system, which are used to obtain the best travelling path (according to user preferences) and critical points, such as intersections in which the user may take the wrong turn. Through this set of default values the Speculative Computation module is used as a mechanism that determines if it is necessary to issue an alert or not. The integration of these two modules are the main contribution of this work. The speculative framework is independent of how the trajectory mining is achieved using the calculated values to ensure the correct travel. A structured reasoning method is provided through the combination of these modules. After preprocessing the data, it is possible to apply different trajectory mining techniques. Our future goal is to determine the most appropriate one for the problem.

In order to better perceive the advantages and disadvantages of CogHelper, a comparison with different systems (presented in Section 2) should be conducted. However this may not be an easy task since CogHelper has an adaptability feature that is not present in other systems. These systems are mainly focused on the user interface (on how the information should be available to the user) and do not adapt the travelling path to user preferences.

As future work, user privacy issues will be considered by taking into account the work developed in [3].

Acknowledgements

This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT - Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013. The work of João Ramos is supported by a doctoral the FCT grant SFRH/BD/89530/2012. The work of Tiago Oliveira is also supported by the FCT grant with the reference SFRH/BD/85291/2012.

References

[1] S Alper and S Raharinirina. Assisitive Technology for Individuals with Disabilities: A Review and Synthesis of the Literature. *Journal of Special Education Technology*, 21(2):47–64, 2006.

[2] Luis Otavio Alvares, Vania Bogorny, Jose Antonio, Jose Antonio Fernandes De Macedo, Bart Moelans, and Stefano Spaccapietra. Dynamic Modeling of Trajectory Patterns using Data Minig and Reverse Engineering. *ER 2007*, pages 149–154, 2007.

[3] Francisco Andrade, José Neves, Paulo Novais, José Machado, and António Abelha. Legal security and credibility in agent based virtual enterprises. In *IFIP Advances in Information and Communication Technology*, volume 186, pages 503–512. Springer New York LLC, 2005.

[4] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. *ACM Sigmod Record*, pages 49–60, 1999.

[5] Stefan Carmien, Melissa Dawe, Gerhard Fischer, Andrew Gorman, Anja Kintsch, and James F Sullivan. Socio-technical environments supporting people with cognitive disabilities using public transportation. *ACM Transactions on Computer-Human Interaction*, 12(2):233–262, 2005.

[6] José Carlos Castillo, Davide Carneiro, Juan Serrano-Cuerda, Paulo Novais, Antonio Fernández-Caballero, and José Neves. A multi-modal approach for activity classification and fall detection. *International Journal of Systems Science*, 45(4):810–824, 2014.

[7] Zhenyu Chen. Mining individual behavior pattern based on significant locations and spatial trajectories. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2012*, pages 540–541, 2012.

[8] Melissa Dawe. Desperately seeking simplicity: how young adults with cognitive disabilities and their families adopt assistive technologies. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 1143–1152. ACM, 2006.

[9] Fraunhofer Portugal. AlzNav. `http://alznav.projects.fraunhofer.pt`, 2012. [Online; accessed 2012-12-10].

[10] Fosca Giannotti, Mirco Nanni, Fabio Pinelli, and Dino Pedreschi. Trajectory Pattern Mining. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 330–339. ACM, 2007.

[11] GraphHopper. Map matching api. `https://graphhopper.com/api/1/docs/map-matching/`, 2015. [Online; accessed 2015-12-12].

[12] Alan L Liu, Harlan Hile, Gaetano Borriello, Henry Kautz, Pat A Brown, Mark Harniss, and Kurt Johnson. Informing the Design of an Automated Wayfinding System for Individuals with Cognitive Impairments. In *Proceedings of Pervasive Health 09*, volume 9, page 8, 2009.

[13] Anna Monreale, Fabio Pinelli, and Roberto Trasarti. WhereNext : a Location Predictor on Trajectory Pattern Mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, pages 637–645, 2009.

[14] João Ramos, Angelo Costa, Paulo Novais, and José Neves. Interactive Guiding and Localization Platform. *International Journal of Artificial Intelligence (IJAI)*, 12(1):63–78, 2014.

[15] João Ramos, Paulo Novais, Ken Satoh, Tiago Oliveira, and José Neves. Speculative Orientation and Tracking System. *International Journal of Artificial Intelligence (IJAI)*, 13(1), 2015.

[16] João Ramos, Tiago Oliveira, Paulo Novais, José Neves, and Ken Satoh. An alert mechanism for orientation systems based on Speculative computation. In *Innovations in Intelligent SysTems and Applications (INISTA), 2015 International Symposium on INnovations in Intelligent SysTems and Applications*, pages 1–8. IEEE, 2015.

[17] João Ramos, Ken Satoh, Paulo Novais, and José Neves. Modelling an Orientation System based on Speculative Computation. In Sigeru Omatu, Hugues Bersini, Juan M. Corchado, Sara Rodríguez, Pawel Pawlewski, and Edgardo Bucciarelli, editors, *Distributed Computing and Artificial Intelligence, 11th International Conference*, volume 290 of *Advances in Intelligent Systems and Computing*, pages 319–326. Springer International Publishing, 2014.

[18] Ken Satoh. Speculative Computation and Abduction for an Autonomous Agent. *IEICE - Trans. Inf. Syst.*, E88-D(9):2031–2038, 2005.

[19] Robert L. Spitzer, Miriam Gibbon, Andrew E. Skodol, and Michael B. First. *DSM-IV casebook: A learning companion to the Diagnostic and Statistical Manual of Mental Disorders.* American Psychiatric Association, 4 edition, 1994.

[20] Yang Ye, Yu Zheng, Yukun Chen, Jianhua Feng, and Xing Xie. Mining individual life pattern based on location history. In *Proceedings - IEEE International Conference on Mobile Data Management*, pages 1–10, 2009.

[21] Y U Zheng. Trajectory Data Mining : An Overview. *ACM Trans. Intell. Syst. Technol.*, 6(3):1–41, 2015.