Where do you roll today?
Trajectory prediction by SpaceRank and Physics Models

Stefano De Sabbata, Stefano Mizzaro, Luca Vassena

Department of Mathematics and Computer Science
University of Udine
Via delle Scienze, 206 – 33100 Udine – Italy
stefano.desabbata@gmail.com, {mizzaro, vassena}@dimi.uniud.it

Abstract: Pre-destination, the prediction of a user's future destination, is recently gaining interest and importance in location-aware, ubiquitous, and mobile computing. An increasing amount of data related to position of people is becoming available because people usually take their mobile devices (phones, smartphones, PDAs, etc.) with them. We propose to mine these data to derive the importance of the single locations in an area of interest, given by either a single user or a community. Then we use the importance of locations as basis for our approach to pre-destination, where well-known physics models (namely gravitation and electrical force) are exploited to estimate users trajectories and future destinations.

Keywords: location-awareness, location importance, physics models, trajectory, destination prevision

1 Introduction

Pre-destination can be defined as the attempt to predict the future location of the user at a given time point. In the location-aware, ubiquitous, and mobile computing communities, pre-destination is recently gaining interest and importance due to actual pervasiveness of mobile devices in conjunction with growing people's need of location-based information systems and services, for example in traffic control, transportation planning, computational advertising, proximity marketing, etc. The location of a user is an important feature for many systems, and it is studied in several respects (Eagle & Pentland 2006). In principle, pre-destination allows to implement more efficient systems, since it allows to set up the destination information environment before the destination is reached. At the same time, pre-destination can be useful in reducing uncertainty when determining users or items locations.

An increasing amount of data related to position of people (for example the usual route taken when driving to work, the usual path followed by people walking in the mountains, etc.) is becoming available because people usually take their mobile devices (phones, smartphones, PDAs, etc.) with them, and these mobile devices can (and indeed they do) leave a record of their past positions. Several technologies allow the generation of this huge amount of data: GSM, UMTS, HSDPA, etc. phones frequently exchange data with the network antennas; Bluetooth and Wi-Fi devices can see and be seen by other devices in close proximity; GPS and Galileo can determine the device position with good accuracy; triangulation and trilateration can be used with all these technologies to provide a more accurate estimate; and so on.

Recently, a large amount of this kind of data has been collected and analyzed, deriving that it is generally more likely that a user will go in a place more related to his/her habits (Gonzalez et al. 2008). Starting from these considerations, we propose to mine the data related to position of people to derive the importance of the single locations in an area of interest, given by either a single user or a community. Then we use the importance of locations as the basis for our approach to pre-destination, where well-known physics models (namely gravitation and electrical force) are exploited to estimate users trajectories and future destinations.

The paper is structured as follows. We first briefly survey related work in Section 2, describing our approach, named *SpaceRank*, in which we show the encoding of users behavior with graphs and the PageRank algorithm, in order to associate importance values to locations (De Sabbata et al. 2008). In Section 3 we present our physics based approach to pre-destination, while in Section 4 we use our approach to analyze experimental data. At the end we present some conclusions and the future work.

2 Related work

New mobile devices and new technologies allow to record the movements of single users. All these records are an interesting data source that can be used to determine past people positions; in turn, mining people positions can allow to derive the importance, or popularity, of places in the real world; and, finally, importance/popularity of locations in the real world can be very useful in order to predict future users locations.

2.1 Give importance to locations

Past user (or users) behavior can be exploited in this respect: generally speaking, it is more likely that a user's position in the future will be a location where he or she has already been (or is used to go to) than a position where he or she has never been (or usually does not go to). Also, if one has to predict the user next position, it is generally more likely that a user will go in a place more related with his/her habits and interests (Gonzalez et al. 2008). So a first step consists in determine the importance of the locations to the user(s).

2.1.1 Classical indexes

In order to define a location importance to a user, it is straightforward to think of three indexes:

- *#visits*: number of visits by the user in the location;
- *avgTime*: average time spent by the user in the location;
- *totTime*: total time spent by the user in the location.

These indexes can be computed on the basis of the data collected during a certain period of time (one day, one week, one month, etc.). This procedure can be tailored either to a single user or to a community: thus, the results will be related to the single user habits or to the community behavior. This approach is followed in several studies (Eagle & Pentland 2006, Ashbrook & Starner 2003, Chan et al. 1998).

If these indexes are considered separately, they just give a partial vision of the user's behavior, and for some applications this could not be enough. For example, if we consider *#visits* only, a location where the user passes by without stopping and a location where the user stops for a long time would be indistinguishable. Similarly, using *avgTime* could put at the same level locations with rare visits and a location with frequent visits. The last index, *totTime*, could be a good importance index since a location should be visited frequently or for long periods to have a high *totTime* value. However, for example, we would like to give different importance to locations having the same *totTime* value, on the basis of one or both of the other two indexes.

One approach could be to use a combination (e.g., a linear combination) of the three indexes, or of some of them. However, this would rise the problem of the choice of both the indexes and the weights to assign to each of them.

2.1.2 SpaceRank

Starting from the limitations of the previous indexes, we have proposed a novel approach to define a location importance to a user. Our approach, named SpaceRank (De Sabbata et al. 2008), is based on the PageRank algorithm.

PageRank (Page et al. 1998) is a well known algorithm for the analysis of links on a hyper-linked set of documents, whose goal is to measure the importance of each document within the set. PageRank assigns a numerical weighting to each element in the set and so represents the likelihood that a person will arrive at that page, randomly clicking. PageRank is used, together with thousands of other features, by search engines to rank the pages retrieved after a user query.

Our aim is to determine each location importance on the basis of geographical properties and users past movements among the locations; past data can concern either a single user (private importance) or a group of users (social importance). We start by concentrating on an area of interest, that we divide into a finite n contiguous sub-areas, defined as *locations*. We suppose that the locations are contiguous and there are no disconnected locations. The number of neighbors for each location is dependent on how the area is divided into locations; in our experiments we use square locations of the same size.

Our approach is based on the linear combination of two $n \times n$ matrix matrixes, where n is the number of locations in the area, according with the equation 1.

$$SpaceRank = (1 - d) \times HabitsMatrix + d \times TransitionMatrix \qquad (1)$$

The *Transition Matrix* encodes the relations of contiguity among locations, represented as a graph (Figure 1a), while the second (*Habits Matrix*) encodes the habits of the user(s) (Figure 1b). The *d* parameter is used to make the results more related to user habits or to geographical properties.

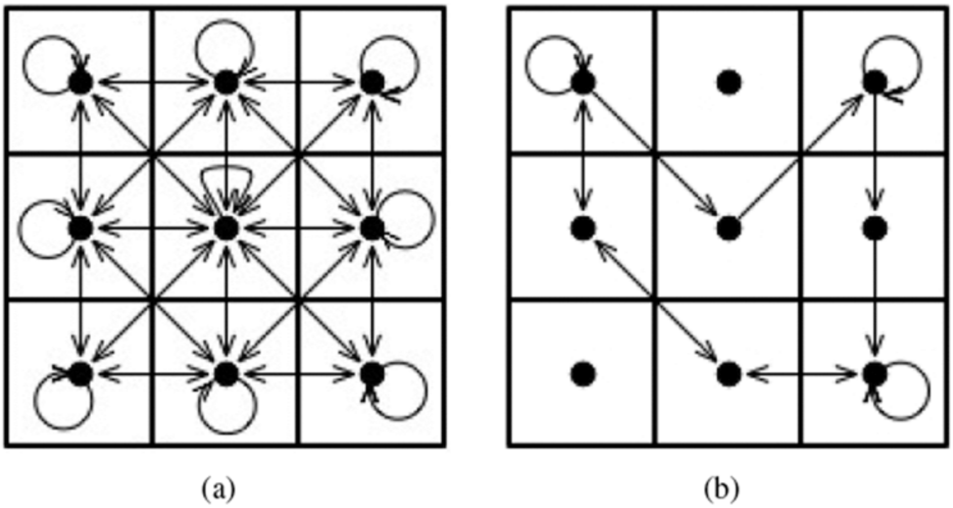


Figure 1: *An example of a base transitions graph (without edges weights) with nine square locations (a). An example of an historical data registry based graph (without edges weights) with nine square locations (b).*

To create the *Habits Matrix* we record users' movements, sampling at regular intervals position and speed. For example if we have recorded temporally subsequent user's position related to the same location, the probability that the user will remain in that location will increase. On the contrary if we have recorded temporally subsequent user's positions related to different locations, the probability of transition between the locations will increase.

Speed is also used to discriminate if the user is moving or if he is standing still in a location. For example, assuming that we have registered two temporally subsequent elements with the same location, $speed > 0$ suggests that user is moving, while $speed = 0$ suggests that the user is standing still in the location. If we consider as permanences (i.e., loops on the same location) only the records with $speed = 0$, the importance of locations where the user has stopped is increased, while the importance of locations where the user has just passed through is decreased. In the same way if we consider only the records with $speed > 0$, the results, obtained with the computation of the importance, will highlight the areas that require a high amount of time to be crossed, reflecting locations with high traffic density because of traffic jams.

2.2 Prevision of the destination

The knowledge of future users locations in the real world can be very useful for several applications. For example, in traffic control (Taylor et al. 2000), in itineraries planning (Bierlairea & Frejinger 2008), in network handover (Zaidi & Mark 2005), and, in general, in location-aware applications (Hazas et al. 2004), knowing which location is the more likely future destination would allow resources optimization, more efficient services, and even more features. Indeed, in the field of context-aware applications, although several factors are taken into account in order to define user's current context, location is usually the main one.

The predestination algorithm is presented in (Krumm & Horvitz 2006) as a procedure that aims at predicting a driver destination point using the history of driver's past destinations and driver's behavior data. More generally, we can define pre-destination as the problem of calculating, given an initial point, direction, and speed, a set of possible destination points with their own probability value.

Mountain, in his Ph.D. thesis (Mountain 2005), presents an extensive study of location-based filters for information retrieval and introduces a simple system to predict the user's next position, speed and heading in order to improve spatial proximity, temporal proximity and speed-heading criteria.

In (Krumm & Horvitz 2006) is proposed a pre-destination approach for a car driver, based on Bayesian inference driven by data about driver's (i.e., the moving object in the space taken into account) previous destinations and behavior, to produce a probabilistic map of possible destinations.

All these proposals are based on a large set of heuristic data. Indeed, so far, pre-destination is mainly an experimental discipline lacking a complete formalization.

3 Trajectory prevision

3.1 Problem formalization and working hypotheses

We can provide a first formalization of the problem as following. We take into account a generic moving object, be it a human walking, cycling, or driving; a car; a bus; an airplane; a train; etc. We refer to the generic moving object as *Pre-Destined Body* (PDB). We assume that it moves on a two dimensional surface, but the extension to the three-dimensional case is straightforward.

Our goal is to determine the probability that the PDB will be at a specific (x,y) location of the surface at some time t in the future, given its position (x_0,y_0) and its movement vector $\vec{v}_0$ at time t_0. At first glance, we can say that we are searching for a function like:

$$p(x_0,y_0,t_0,\vec{v}_0,x,y,t). \qquad (2)$$

So, that function should compute the probability that (x,y) will be PDB position at time t.

However, people normally do not move around without being attracted by some places. So it seems obvious that the placement of interesting objects will affect in some way the movement of the PDB: on the surface are placed n *Points Of Interest* (POIs), that could be houses, offices, shops, cities, etc.

So, the function we are looking for could be something like

$$p(x_0,y_0,t_0,\vec{v}_0,x,y,t,[k_1,\ldots,k_n]), \qquad (3)$$

where $[k_1,\ldots,k_n]$ is a list of k_i elements that represent all the information about singles POIs, such as their position (x_i,y_i). Or, in an equivalent way, we can take $[k_1,\ldots,k_n]$ as fixed and define $p(x,y,t)$ in the space shaped by $\left[k_1,\ldots,k_n\right]$:

$$p_{[k_1,\ldots,k_n]}\left(x_0,y_0,t_0,\vec{v}_0,x,y,t\right). \qquad (4)$$

3.2 Physics metaphors

The aim of an abstract model is to simplify the analysis of the problems and provide a common basis for the interpretation of different phenomena. In our opinion, a suitable way to do that is to rely on a metaphor derived from real world or widely known theories. In this paper we try to show how concepts and ideas drawn from physics can provide both intuitive metaphors and well established theories, that can be useful to understand and formalize in a more precise way the above defined pre-destination problem. We first propose our basic model, and then present several improvements to make it less abstract and more related to the real world.

3.2.1 Classic gravitation

If you are walking around in a museum, some items will attract you, and some items will not capture your attention. So you are not so different from a moving object in a gravitational system, or an electric charge in an electrostatic field directed towards a specific point. The basic intuition behind our approach is to use the well-known geometrical theory of gravitation, published by Albert Einstein as the Theory of General Relativity, as a metaphor for general pre-destination scenarios.

In this metaphor, the PDB is like a ball rolling with no friction and negligible mass and diameter (as a point particle) on the surface. POIs are static bodies, with a mass and a diameter that encode some properties of the object as, for example, the number of previous visits, the number of people that are currently visiting it, or the average of time spent in that place. So, POIs act like a planet in a gravitation system and, because of their mass and diameter, create a distortion of the surface, that can change the trajectory of the PDB as it is sketched in Figure 2. More important objects have a distortion that is both wider and deeper.

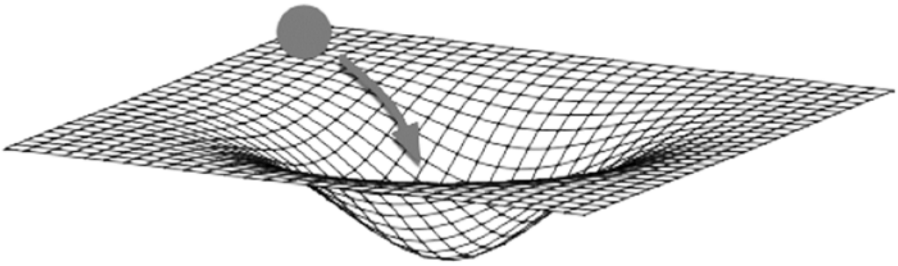


Figure 2: *A POI changing the shape of the surface and attracting the PDB*

3.2.2 Attraction and repulsion: from gravitation to electrical force

While the gravitational force is an attractive force, the electrical or Coulomb one can be both attractive and repulsive, depending on the electric charges. From this point of view, a pre-destination model based on Coulomb's law is more general and complete than the classic gravitational model. In the gravitational model, a POI can just attract a PDB as it is important for it, while in the electrical model a POI can also repulse PDBs, as in Figure 3. To illustrate this situation, we can think about a supermarket: a user could change his path if it leads to a too crowded lane.

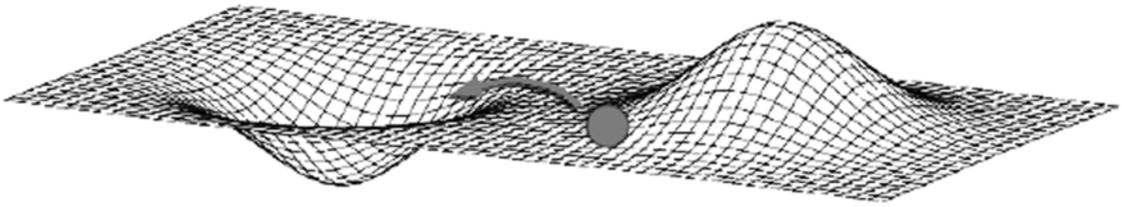


Figure 3: *A PDB in a Coulomb model*

Thus, the surface presents depressions and peaks, of different height and depth. Sometimes repulsion is certain (a man cannot walk through a wall; a metro car must move inside a tunnel network; etc.). Attraction can be certain too in some – indeed more specific – cases (e.g., a train moving at high speed and thus having a high inertia; etc.). Thus peaks are sometimes walls of infinite height and depressions are sometimes holes of infinite depth.

3.2.3 More objects in movement

The previously described model presents a limitation: it is a deterministic model. Given the initial (time t_0) parameters of position, direction and speed, there will be only one point where the PDB can be at a time $t > t_0$. Also, the (x, y) position is determined with no uncertainty, which is usually not true in real world applications and situations (e.g., as in GPS systems), and is unrealistic, since usually the future cannot be predicted with certainty. In practice, the probability distribution degenerates to a Dirac's Delta function, having a 1 value in the (x, y) point where the ball is at time t, and a zero value in all other points. In the following we will improve our model in order to overcome these limitations.

A first attempt could be to consider, in place of a single ball with precise initial position, a ball that can be in one out of a set of initial points. Then the ball can be let to roll, and this "experiment" can be repeated for every point in the initial points set. The final destinations will be in general different, and the different experiments results can be averaged in an appropriate way.

This could work, but is rather inconvenient. A simple improvement is to consider a set of balls, rolling contemporarily, with no friction, and no interferences among the balls. In general, the differences in the initial points could lead, after some time, to locations that could be different, even to a great extent (see Fig. 4). Using this simple model we can assume that the probability for PDB is proportional to the number of balls at (x, y) at time t.

To represent this situation we:

- associate to a user more PDBs arranged uniformly on the region representing the hypothetic user location;
- assign to each PDB a probability value, representing the probability that the PDB reflects the real user position;
- compute the prevision for each PDB.

In this way, at each time, the user position is represented by a probability distribution associated to different locations, as sketched in Figure 4.

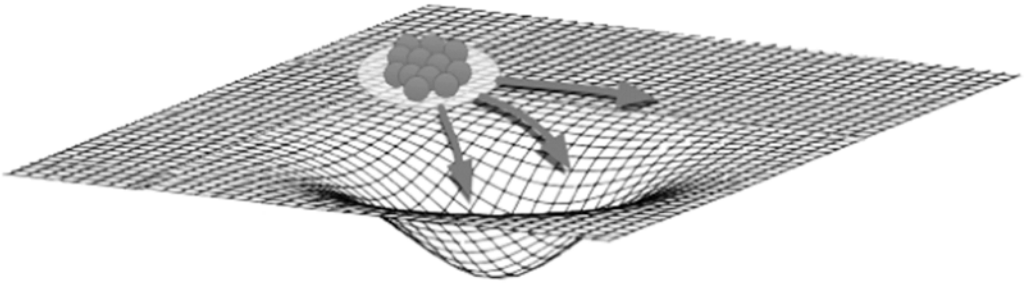


Figure 4: *More objects in movement*

3.3 Pre-destination algorithm

3.3.1 From importance to acceleration

We now propose our approach to pre-destination, named ARDA (A Rolling to Destination Algorithm). We start from the results obtained analyzing user's habits with our SpaceRank algorithm, although this is not a strict choice as our algorithm is flexible and independent from how the importance values have been determined: the only requirement is that the importance values are stored in a matrix representing the locations grid. Let's call this matrix M_{imp}.

Having adopted the electrical physics model, we need a matrix M_{pot}, representing the electrical potential energy in each location. In this model, a PDB will be attracted by lower potential areas, so, in order to have the PDB attracted by more important locations, given that more important locations have higher values in M_{imp} matrix, we need to use opposite importance values in our physics model. So, starting from M_{imp} we generate a matrix with opposite values M_{pot}, representing the locations potentials. We use the function $f(x) = -x$ for simplicity, but any monotonically decreasing function could be used. Then we compute the matrix M_{acc} of accelerations depending on the potentials in the area of interest. Each cell of M_{acc} contains the acceleration vector related to the location represented by the matrix cell. M_{acc} is computed using the gradient:

$$M_{acc} = -\vec{\nabla}\left(M_{pot}\right). \qquad (5)$$

3.3.2 Trajectory computation

We can now compute the future trajectory on the basis of simple physics laws. We start from:

1. the user position (x_0, y_0) at time t_0;
2. the movement vector $\vec{v}_0$ at time t_0;
3. the acceleration matrix M_{acc};
4. a function $acc(x, y)$ that returns the acceleration value referred to the location (x, y). In this function an attenuation factor γ_{acc} is used to adapt vectors magnitude to the magnitudes in the physic simulation. In fact physics simulation's data on user position and moving vector are related to real world values fitted to location's size in the grid, whereas acceleration vectors' magnitude is related to locations' importance estimation values, which are not related to real world sizes;

5. a period T_c to use in user state's next sampling simulation.

We use the uniformly accelerated motion law to simulate the itinerary of the user and therefore to predict the future user position (x_1, y_1) and movement $\vec{v}_1$, at time $t > t_0$:

$$(x_1, y_1) = (x_0, y_0) + \vec{v}_0 T_c + acc(x_0, y_0) T_c^2 \tag{6}$$

$$\vec{v}_1 = \vec{v}_0 + acc(x_0, y_0) T_c \ . \tag{7}$$

If we consider a probability distribution as initial position (as described in section 3.2.3), we have to apply the described procedure to each PDB and, for each location in the area of interest, the probability to be the user position is equal to the sum of the probabilities assigned by the computation of the procedure on PDBs.

3.3.3 Holding factor

We then introduce a "holding factor" as a procedure to slow down user's speed on the basis of the importance of the location the user is in.

The holding factor can be interpreted as the probability that, given the importance of the location the user is in, the user will slow down and stop in that location. The holding factor is simulated computing the movement vector as follow:

$$\vec{v}_{j+1} = \frac{\phi}{\gamma_\phi} \vec{v}_j , \tag{8}$$

where, given the importance value l_{imp} of the location where the PDB is in, $\phi = 1 - l_{imp}$ is the value associated to the holding factor and γ_ϕ is the value used to attenuate the same factor (to adapt the importance value to the magnitudes used to encode the information related to the real world).

3.3.4 Considerations

The described algorithm uses as a main information source the matrix M_{imp} with the importance values computed by SpaceRank. However several other different approaches can be used to define the matrix of importance. For example, we can integrate SpaceRank with information in the user calendar, or we can use the user history to improve the computation of the importance. We can also take into account social information together with the user profile: for example if a location is important for most students and I am a student, maybe that location is important for me too.

From this point of view the model we propose is flexible and we can compute the importance matrix on the basis of different sources:

$$M_{imp} = \alpha_1 M_{source1} + \alpha_2 M_{source2} + \ldots \tag{9}$$

Moreover our ARDA algorithm can be used both for the trajectory prediction in a short time (next minutes) and for the final route destination prediction. The main problem of our algorithm is represented by the attraction of the starting location, as in most cases it is an important location. This location applies an attractive force to the PDB that will affect all the previsions related to the first part of the route.

The solution we have adopted is to smooth the starting location attraction by means of a monotonically decreasing function related to the distance from the starting point. As there is a probability < 10% that a route will last less then 4 minutes (Krumm & Horvitz 2006), we apply the smooth function to all the locations reachable in 4 minutes.

4 Preliminary evaluation

In this section, some preliminary experiments are presented; the aim of these experiments is to test our ARDA algorithm and verify its correctness and effectiveness.

4.1 Experimental settings

In these experiments, the log data concern a typical week of one of the authors of this paper, collected manually. The relevant area, Udine town and his surroundings, is divided into 9 × 16 squared locations, each of them having 700 meters as side length. The ARDA algorithm has been developed using Python and the following modules: PyXML for XML processing, RPy for statistics computation, SciPy and NumPy for matrixes computation.

The aim of these experiments is to test our ARDA algorithm and verify its correctness and effectiveness; future experiments will have more users, more itineraries, and a more detailed location subdivision.

We started by computing the locations importance using SpaceRank, with different parameter values; the result is shown in Figure 5.

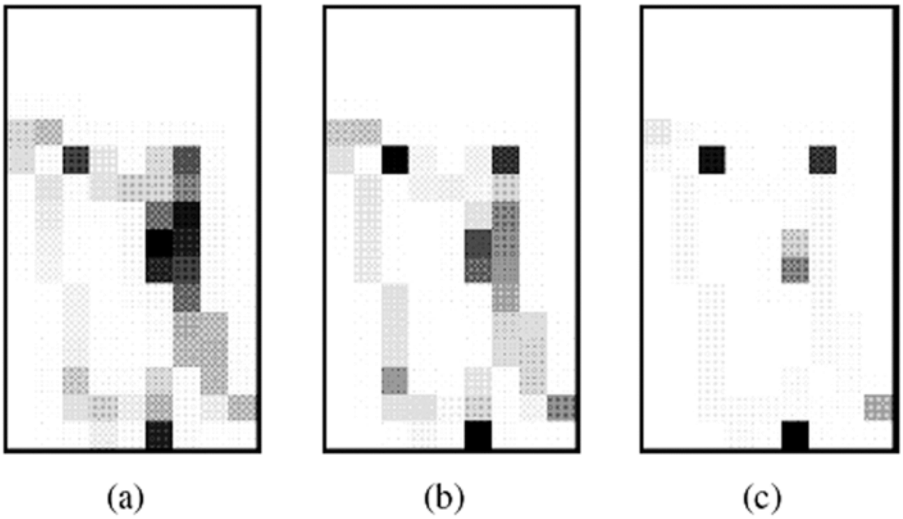


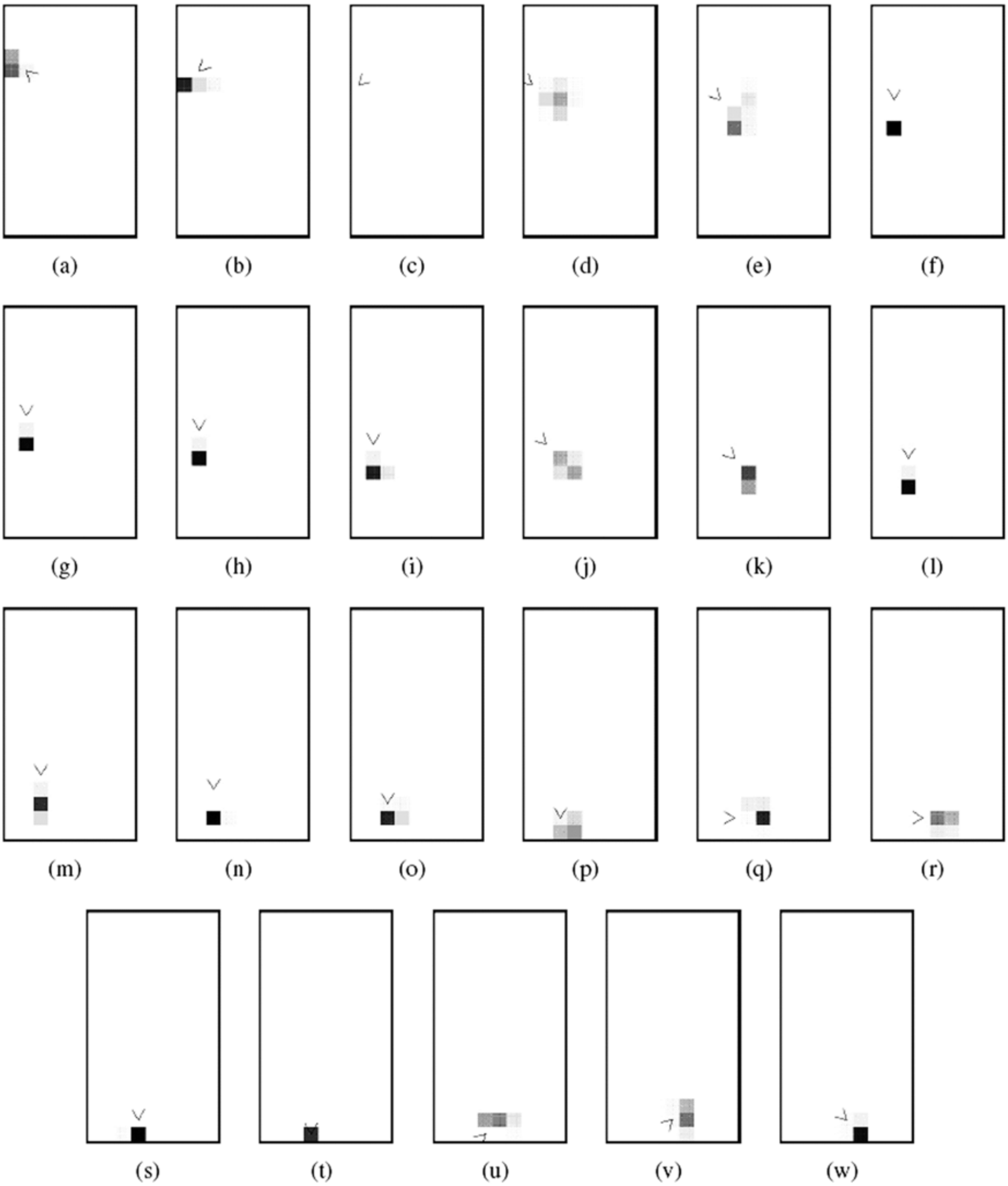
Figure 5: *Importance computed by SpaceRank with* $d = 0.15$ *(a), with* $d = 0.05$ *(b) and giving more importance to locations where the user stops (c).*

4.2 Short time prediction

In this first experiment we have applied our ARDA algorithm to predict the user location in the next minute. This computation has been applied to 23 samples of user position and moving vector taken every 20 – 40 seconds (in order to avoid subsequent equal previsions) during the route from university to author's home.

In the SpaceRank equation we have used $d = 0.15$ (Figure 5a): this means that we consider not only the locations where the user has stopped, but also the locations the user has passed through. A high importance location (darker squares in the location, in the bottom part of the figure) is the home of the user; another high importance location is the University (in the top left hand side), where the user stays about 9 hours per day 5 days per week. The other high importance locations have been visited for hours by the user. The lighter gray level locations are the roads where the user drives usually. White squares locations have never been visited.

On the basis of position and speed values obtained from the real data, we have experimentally seen that the most suitable values for the parameter in the ARDA algorithm are $\gamma_{acc} = 0.01$ and $\gamma_{\phi} = 0.3$.



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

(m) (n) (o) (p) (q) (r)

(s) (t) (u) (v) (w)

Figure 6: *Results obtained predicting the user position in the next minute.*

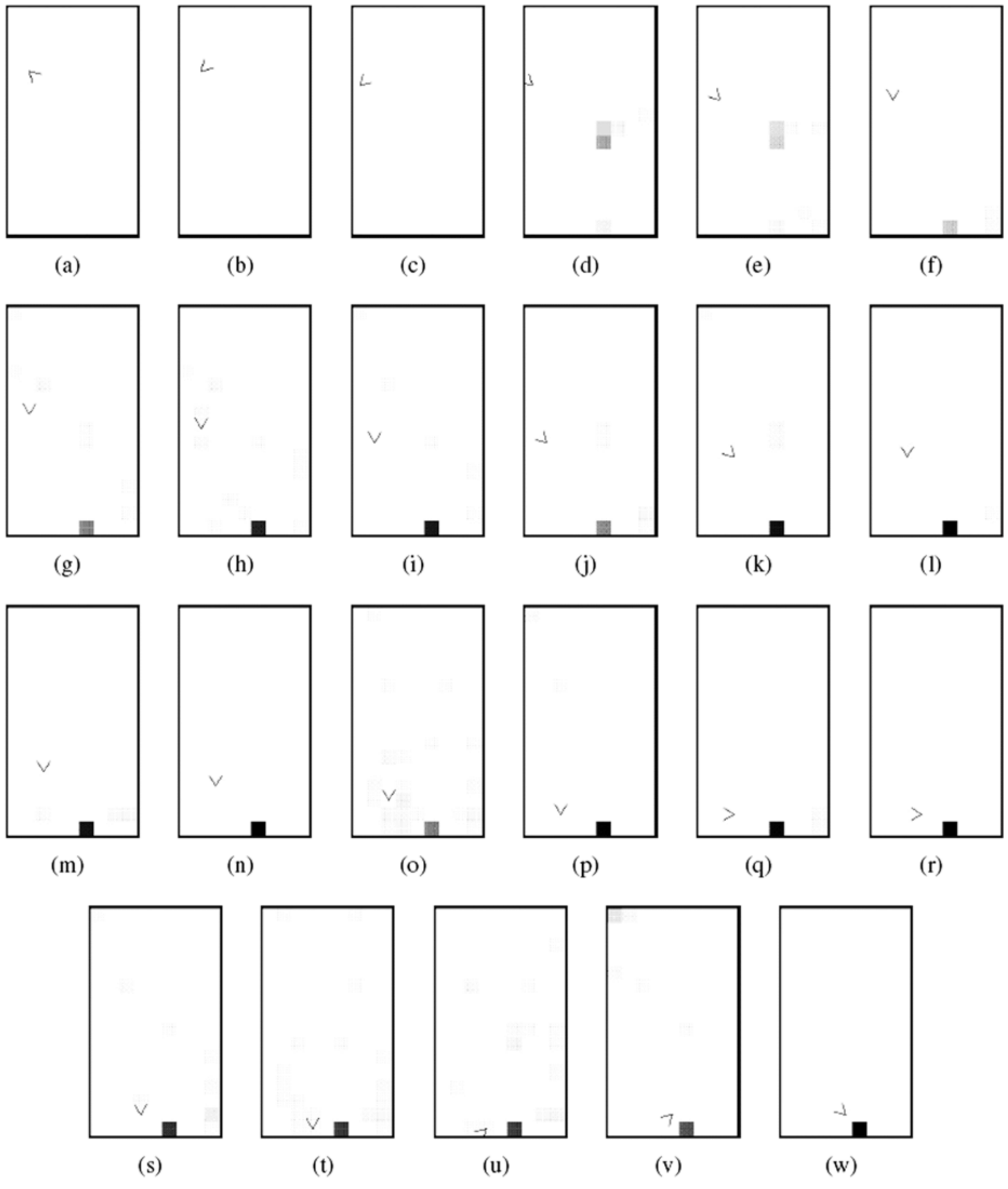


Figure 7: *Results obtained predicting the user final destination.*

In Figure 6 the results of this experiment are shown. Each image shows one of the 23 prevision computations:

each square represents a location, the location color is related to the assigned probability (a darker color means a higher probability that it will represent the user future position) and the little arrow represents the actual position and movement direction of the user.

With a “to 60 seconds” prevision it is possible to obtain a satisfactory prediction of the user’s position in the next 40 − 80 seconds. However because of the granularity chosen in space subdivision, we have observed that it is difficult to have a prediction with accurate timings. A more detailed space subdivision would have signified a more detailed knowledge of the road shape and therefore a more detailed prediction.

4.3 Destination prediction

In this experiment we have applied our ARDA algorithm to a set of positions and movement vectors recorded along the way from university (location (5, 0)) to home (location (2, 10)). The aim of this experiment is to predict the final destination of the user, so the computation stops only when the speed of the user in the simulation is 0.

We have computed the locations importance using SpaceRank with $d = 0.05$, therefore considering all location where user has been (Figure 5b). On the basis of position and speed values obtained from the real data, we have experimentally seen that the most suitable values for the parameter in the ARDA algorithm are $\gamma_{acc} = 0.05$ and $\gamma_{\phi} = 0.3$. As the starting location is important, we introduced a factor to smooth the starting location attraction, as described in Section 3.3.4.

In Figure 7 the results of this second experiment are shown. The images have to be interpreted as in Figure 6; however while in Figure 6 the user position in the next minute is predicted, in Figure 7 the user final destination is predicted.

In general, especially during the first part of the itinerary, we have noticed a strong dependence between the user direction in the instant of the prediction and the obtained prediction result. This is due to the SpaceRank algorithm that assigns to the location where the user moves through importance values very near to the values assigned to the locations where the users stops. In this case, with a low γ_{acc} value there is not enough force to attract the user towards the final destination, and with a high γ_{acc} value the prediction is disturbed by the accelerations related to the locations in the itinerary.

To overcome this limitation, we have performed a further experiment, with a different importance index. In this case, we assign more importance to locations where the user usually stops (*speed* =0), decreasing the values related to the locations where the user just passes through (Figure 5c). This version of the algorithm improves the prediction of the final destination. In particular we can notice that even in the first part of the itinerary, the algorithm assigns high probability values to the location of effective destination. In particular it returns good results, comparable to the other algorithms in literature. Moreover, if the final destination is an important location far from other important locations, our algorithm assigns it a high probability; on the contrary if the final destination is an important location near other important locations, our algorithm tends to equally distribute the probability among these locations.

5 Conclusions and future work

We have emphasized how the prediction of a user’s future location and destination is gaining importance due to actual pervasiveness of mobile devices in conjunction with growing people’s need of location-based information systems and services. Moving from this point, we have proposed an algorithm to predict future user locations. The algorithm has two steps: (i) computation of the importance of the locations, based on geographic properties and user habits; and (ii) prediction of future locations, based on physics models. Our approach is different from the existing solutions, it is intuitively reasonable and, accordingly to preliminary experimental results, it seems effective.

In the future we plan to compare our algorithms with other existing approaches both for importance estimation and for trajectories previsions. Moreover, we intend to study the problem of understanding human behaviors and habits at broader and deeper levels. We plan to mine location-based logs and discover knowledge in these data, in order to better estimate places importance and trajectories not only in real world but also in virtual worlds like the web graph, search engines networks, social networks, and online role play games. For instance, interesting problems would be destination prediction in SecondLife, in Web browsing, in search engine querying, etc.

Our future studies will also focus on what is "importance", how "global importance" is different from "personal importance", how we can estimate places "importance" and how we can use these information in the development of novel context-aware applications both for mobile devices and networking tools.

Finally, we intend to improve our trajectories prediction algorithm both by including alternative importance information sources, such as personal calendar or friends position, and by considering others physics' fields, such as weather forecast, in search of better approaches and models.

References

Ashbrook, D. & Starner, T. (2003), 'Using GPS to learn significant locations and predict movement across multiple users', *Personal and Ubiquitous Computing* 7(5), 275–286.

Bierlairea, M. & Frejinger, E. (2008), 'Route choice modeling with network-free data', *Transportation Research Part C: Emerging Technologies* 16, 187–198.

Chan, J., Zhou, S. & Seneviratne, A. (1998), A QoS adaptive mobility prediction scheme for wireless networks, *in* 'Global Telecommunications Conference, 1998. GLOBECOM 98. The Bridge to Global Integration. IEEE', Vol. 3, pp. 1414–1419.

De Sabbata, S., Mizzaro, S. & Vassena, L. (2008), SpaceRank: Using PageRank to estimate location importance, *in* 'Proceedings of ECAI '08 Workshop on Mining Social Data (MSoDa '08)', pp. 1–5, University of Patras, Greece.

Eagle, N. & Pentland, A. S. (2006), 'Reality mining: sensing complex social systems', *Personal and Ubiquitous Computing* 10(4), 255–268.

Gonzalez, M. C., Hidalgo, C. A. & Barabasi, A.-L. (2008), 'Understanding individual human mobility patterns', *Nature* 435(7196), 799–782.

Hazas, M., Scott, J. & Krumm, J. (2004), 'Location-aware computing comes of age', *Computer* 37(2), 95–97.

Krumm, J. & Horvitz, E. (2006), Predestination: Inferring destinations from partial trajectories, *in* 'Lecture Notes in Computer Science', Vol. 4206, Springer, pp. 243–260.

Mountain, D. M. (2005), Exploring mobile trajectories: An investigation of individual spatial behaviour and geographic filters for information retrieval, PhD thesis, City University.

Page, L., Brin, S., Motwani, R. & Winograd, T. (1998), The PageRank citation ranking: Bringing order to the web, Technical report, Stanford Digital Library Technologies Project.

Taylor, M. A. P., Woolley, J. E. & Zito, R. (2000), 'Integration of the global positioning system and geographical information systems for traffic congestion studies', *Transportation Research Part C: Emerging Technologies* 8, 257–285.

Zaidi, Z. R. & Mark, B. L. (2005), 'Real-time mobility tracking algorithms for cellular networks based on kalman filtering', *IEEE Transactions on Mobile Computing* 4(2), 195–208.