Finding Interesting Places and Characteristic Patterns in Spatio-Temporal Trajectories

Udo Feuerhake, Colin Kuntzsch, Monika Sester

Institute of Cartography and Geoinformatics,
Leibniz Universität Hannover, Germany

Abstract. Trajectory data often includes knowledge about the movement and the behavior of individuals, which is useful for analyzing problems in domains like animal migration or security. In this paper we present an approach to identify interesting places and determine unusual behavior of individuals from large amounts of trajectory data.

Keywords. Spatial, Data Mining, Pattern, Places, Segmentation, Extraction, Algorithms, Incremental

1. Introduction

Understanding of space-time-patterns is relevant for many problems, e.g. animal migration, traffic analysis, security. This paper describes first stages of a framework for the interpretation of trajectory data with respect to specific patterns or dominant structures. There are several requirements concerning the interpretation of such space-time-trajectories, which are also application dependent: The trajectories can have different sampling rates, they can contain additional attributes (besides position); they can be generated by the same individual or several individuals, which may or not be known beforehand. The goal can be to identify certain patterns, which can be discerned into individual patterns of the trajectory itself and group patterns of several trajectories. A classification of space-time patterns is described by Dodge, et al. (2008). Concerning the first issue, the way the space is traversed is relevant, e.g. in terms of straight lines, circles, zigzag; then methods described by Buchin et al. (2009) can be applied. Concerning the group patterns, dynamic patterns like flocks, and also static patterns like convergence or encounter (see e.g. Laube et al. (2008)) can be identified. Furthermore, also the issue of identifying regions of common space usage can be identified, a sort of linearly extended encounter. This latter pattern can also be used to identify the underlying path (network) of the moving objects (e.g. Krumm, et al. (2009)).

In the paper, we will concentrate on a scenario with different individuals traveling through space and time. We are first of all looking for encounters in the sense of places, where the individuals recurrently appear. To this end, an incremental approach is presented. After the places of interest are identified, a graph can be constructed, consisting of the interesting places as nodes and the connecting track segments as edges. Further analysis is conducted on these segments, with respect to movement type, as well as the possibility of aggregating nearby traces. Finally, general information about the connections in the graph is derived, which also include probabilities of paths through the network.

In the following, a brief description of the algorithms is given, together with illustrative examples. The paper concludes with a summary and outlook on the next steps.

2. Motivation

2.1. Context of Problem

This paper deals with the evaluation of trajectory data in an observation scenario. There the main focus lies on the identification and evaluation of movement patterns to detect critical behavior of observed individuals. Considering the fact that the developed technique shall operate on a low-performance system, it has to work efficiently.

2.2. Problem Definition and Description

A major challenge of the project is to define critical behavior. Due to the fact the evaluation is based on trajectory data, the behavior itself is represented by the movement of an individual. Therefore critical actions are directly related to critical movement. Certainly, there are various criteria characterizing movement as critical. These criteria heavily depend on the spatio-temporal context, the individual is moving in. Thus, when looking for uncommon behavior, we are looking for uncommon trajectories in the data. Common trajectories are identified by looking for clusters of similar locations and paths. Deviations from these clusters are considered as abnormal. However, they do not have to be necessarily critical. After the detection of the requested movements further decisions have to be made to classify an abnormal one as critical. Critical has then to be defined within the context of an application.

In summary we are looking for a method that separates common from special movements according to their spatio-temporal context.

2.3. Related Work

This is not the first approach structuring and evaluating trajectory data in a spatial context. Ashbrook and Starner (2003) describe a way to learn significant locations and make predictions from GPS data. They structure the existing data by finding common places, where people stay for certain time. After that they use a k-mean-similar clustering to merge the found places, to reduce their data to an essential minimum. For each of those resulting clusters, called locations, a Markov model is created, which allows predicting the people's next target.

Makris and Ellis (2001) have worked on identifying frequently used paths from video scenes. Thereby, they handle the spatial relationships among the trajectories by generating a graph from people's appearing- (entry nodes) and disappearing-points (exit-nodes) and trajectory junctions the scene. Node-usage statistics provide a measure about the most probable exit-node, so the point, the individual leaves the scene.

Another approach is presented by Baiget and Sommerlade (2008). They want to find trajectory prototypes to estimate subsequent trajectory shapes. To this end, they cluster the already obtained trajectories by their first and last points. After that, they create a trajectory prototype for each of those clusters by combining the contained segments. Those prototypes are used for predicting the most probable shape of following trajectories.

Kang et al. (2005) developed an algorithm for extracting significant places from a trace of coordinates. Instead of using GPS, they use WiFi to collect users' locations. To extract the interesting places from the location data, they suggest a time-based clustering, which relies on a distance and a time threshold.

2.4. Own Approach

Since the behavior of an individual depends on its spatial-temporal context, our problem also demands a spatial structuring of the existing trajectory data. Otherwise we would not be able to compare and interpret the movements. As in some of the related work, we have also decided to extract attractive places. But, instead of clustering any locations found by longer stays of individuals, we identify interesting places in an incremental way by counting their visits by individuals. These places will later be used to derive clustered logical segments from the trajectory data. So the segments of within each cluster will share their own spatial context.

After having clustered the trajectory segments we are able to evaluate the segments within the same environmental background. Subsequently, the aggregated trajectory segments will be clustered a second time using domain-specific parameters and analyzed with respect to their internal structure.

Thus, we present a three step-approach:

1. Extraction of attractive places
2. Segmentation and clustering of trajectory data based on the found places
3. Evaluation of segments within clusters of semantic trajectory segments

Compared to the related work we mentioned above, the main difference, next to the way of finding attractive places, is the sequence of single steps applied for reaching our goal.

3. A Three Step Approach

3.1. Requirements, Assumptions and Definitions

There are a few prerequisites and basic assumptions of our approach. The algorithm needs a sufficiently large amount of input data with a nearly constant, but quite high sampling rate. The data augmentation of our algorithm is based on the Adrienkos' basic concepts of movement data. So a trajectory $T = \{TP_0, .., TP_m\}$ is defined as a sequence of m measured tracking points $TP = (x, y, z, t)$, which contain values for space and time Andrienko et al. (2008).

We define a place to be attractive, if it has been visited several times by one or more individuals. Therefore, we define a necessary parameter n, the visit count of a place, for separating candidate places, visited n-1 times, from attractive ones. Reasonable values for n basically depend on the phenomenon to be analyzed and on the spatial density of data. The higher the density the higher n should be.

Furthermore, we define the size of a place's geometry. The smaller the size is set, the more places and clusters may be found.

Finally, we assume that individuals are getting slower or even stop at attractive places. In general, stopping can be identified by analyzing the velocity. In this work we use a fixed velocity threshold v between two consecutive tracking points. If additional knowledge about the objects or the scene is known which influence stopping, they can also be included.

A more detailed evaluation of the influences of the parameters will be presented together with the discussion of the results.

3.2. Step 1: Extraction of Attractive Place

In generally, candidate places are found by examining movement data and stops of individuals. Those candidates will upgrade to “attractive places” if a threshold for the number of visits (n) is reached. Since our search for candidates works on every single tracked movement, the algorithm is operating incrementally. So it works at the runtime, which is important when using it in a surveillance system as described above. In the following paragraph we explain the extraction of attractive places in detail.

For each pair of consecutive observations we decide, whether the observed individual/object has moved significantly. In this work we use the observed velocity, calculated from the travelled distance between two samples with known timestamps. This approach can be easily adapted for domains with high sampling-frequency by aggregation of more than two consecutive observations. If the calculated velocity fulfills certain stop-criteria, the observed movement M is interpreted as a stop along the trajectory. M’s center O is tested for containment within an existing place. If this check fails, i.e. there is no existing place containing the movement’s location, a new candidate place (C) is created and added to the set. In the other case the found place’s (P’s) center will be adjusted. This correction consists of moving the current center towards O. The new center will be the mean of all previous movement centers contributing to P. Further, P’s visit count is increased. If the count reaches the predefined threshold n, P will be considered an attractive place and put into the corresponding set. The following pseudo code describes the first step of the algorithm.

```
Places = Ø, AttractivePlaces = Ø, Place P, Movement M, Location L
FOREACH tracked movement
  M = new tracked movemen
  calculating parameters of M
  O = center of M
  IF M satisfying stop-criteria
      FOREACH P ϵ Places
          IF O inside P
              Increase visit count of P
              Correct center of P
              IF visit count == n
                  Upgrade P to attractive place
                  AttractivePlaces = AttractivePlaces ∩ P
              END IF
          ELSE
              Create new candidate place C
              Places = Places ∩ C
          END IF
      END FOR
  END IF
END FOR
```

3.3. Step 2: Segmentation and Clustering of Trajectory Data

In our next step the clustering requires a segmentation of the existing trajectories. Therefore the attractive places which have been found in the previous step are used. These are considered as the start and the end points of the segments, respectively.

The segmentation algorithm iterates over the tracking points of each trajectory and checks, if the current tracking point is contained inside any of the previously identified places (during the segmentation not only stops are considered, but also motion passing through a place). Whenever a trajectory point passes a place, the previous segment, if there is one, is closed and stored separately. A new segment starting at the previous trajectory segment's end place is initialized. Trajectory points not inside any of the place models are added to the current segment. Segments at the start or end of trajectories not starting or ending at place models are discarded. The result of the algorithm is a set of trajectory segments, each associated with a pair of places it starts or ends at.

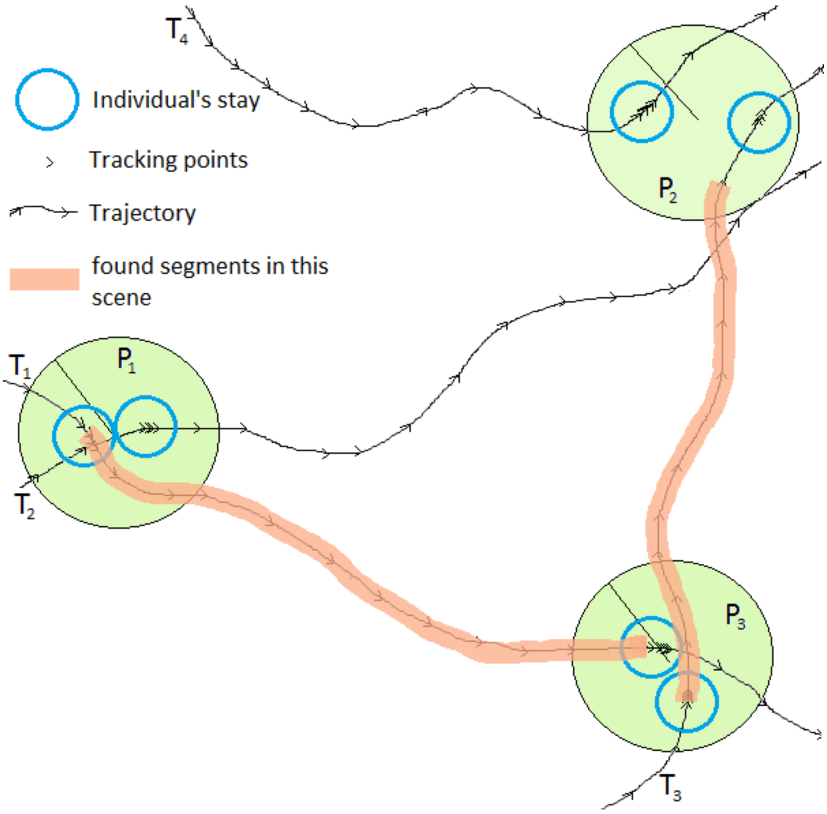


Figure 1: Illustration of trajectory segmentation

```
Trajectories = {T0..TN}, Trajectory T, TrackingPoint TP
AttractivePlaces places = {P0..PM}, AttractivePlace P, AttractivePlace Pstart,
Segments = Ø, Segment S = Ø
FOR EACH T ϵ Trajectories
  FOR EACH TP ϵ T
      FOR EACH P ϵ places
          IF TP inside P
              IF S ≠ Ø
                  S = S ∩ TP
                  IF P ≠ Pstart OR T has been outside of Pstart
                      Segments = Segments ∩ S
                      S = Ø
                  END IF
              ELSE
                  Create new Segment S = { }
                  S = S ∩ TP
              END IF
          ELSE
              IF S ≠ Ø
                  S = S ∩ TP
              END IF
          END IF
      END FOR
  END FOR
  S = Ø
END FOR
```

After the segmentation, we cluster the results. To this end, we use the places again. The clustering features are the start and end places of the trajectory parts, so that every cluster is defined by a pair of places. For the example in Figure 1, there will be every combination of the places P_1, P_2 and P_3, with consideration of their order. The matrix in Figure 2 gives an impression, how the result looks like for the simple schematic example in Figure 1.

Each entry of that matrix represents the number of segments belonging to one cluster. There can be entries also on the diagonal, which means that there are loops, so segments start and end at the same place.

	P1	**P2**	**P3**
P1	0	0	1
P2	0	0	0
P3	0	1	0

Figure 2: The result of the clustering showed as a matrix

This matrix can also be regarded as a non-complete graph, where nodes are represented by the places (P_1, P_2, P_3) and edges by entries greater than 0. This fact does not really matter for the current state of work, but it may be useful for future aspects.

3.4. Step 3: Evaluation of Segments within Clusters

The segmentation presented above leads to a reduction of complexity of the collected data. Instead of one single trajectory per observed individual, we are now able to operate on trips between places. Those trips can be utilized for generating a model of typical migration behavior between places. In this work we present an approach based on clustering of spatial and geometric attributes generated by a single trajectory analysis by means of different strategies of getting from place A to place B. Depending on the domain of observation, different trajectory parameters play more or less important roles in distinguishing those place-crossing strategies and mapping them to semantic categories, e.g. spatial proximity and similar shape may implicate the use of similar routes in a street network, while the same properties would be less useful in domains without spatial restrictions on movement.

Those strategies among trips can be identified utilizing prior domain knowledge to preselect trajectory parameters used for clustering like

- spatial parameters
    - location
- trajectory geometry parameters, e.g.
    - shape
    - curvature
    - sinuosity
- temporal parameters
    - speed
    - time (of day)

4. Experiments and Parameter Evaluation

The first step of our algorithm, the places extraction step, requires three parameters. Reasonable values for the latter depend on the examined scenario. They have to be adjusted to the trajectory density and sampling rate of the data. For that purpose, we use a scenario examplarily to show the influences of the parameters to the resulting number of candidate and attractive places.

A large data set contains several trajectories of animals moving in an area of approx. 100 x 100 km, cf. Figure 3.

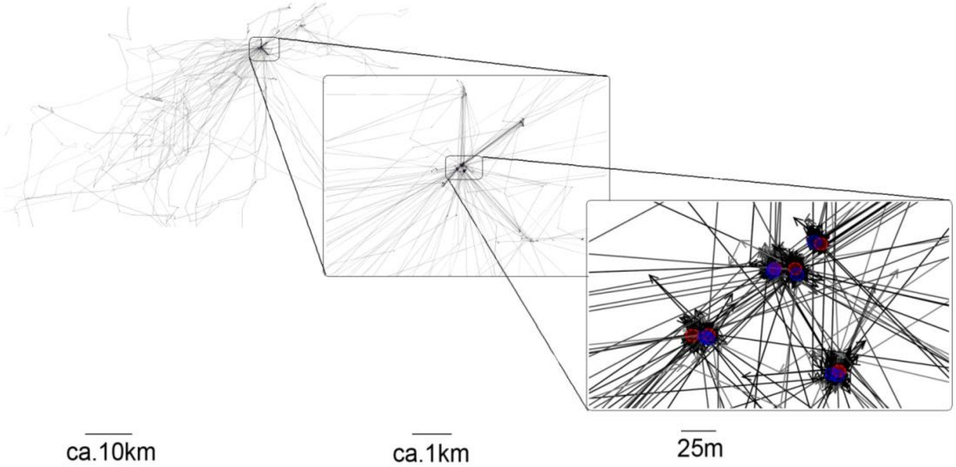


Figure 3: A part of the data set and the places we were looking for showed in three different scales

The parameter settings for the examination of 66149 tracking points are: n=2; r=5m and v=0.1m/s. The data set also contained ground truth in terms of known attractive places, which the individuals often visit. We use this list to verify our results. Our algorithm found more places than given in this list, but it did include the ground truth places as well. An example for the result is showed in Figure 4. There are small deviations, about 5-10m, between the given blue places and the red calculated ones, which mainly can be explained by the inaccuracy of GPS.

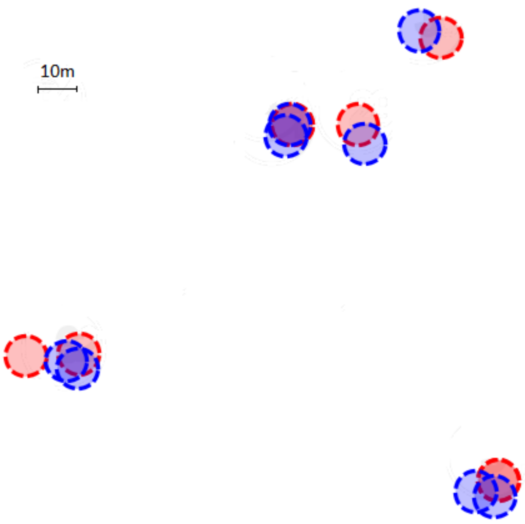


Figure 4: Comparison between blue given places and red determined places

In Table 1 experiments with different parameter settings are listed. We made three series of measurements varying one parameter each time.

#	Parameter setting			Identified places		Remark
	n[-]	r[m]	v[m/s]	Candidates	Attractives	
1	2	5	0.5	5201	356	
2	2	5	1	7969	373	Changing the velocity threshold
3	2	5	2	9939	414	
4	2	5	5	16507	430	
5	2	5	10	29103	458	
6	2	5	20	36346	465	
7	2	5	30	36354	465	
8	2	5	50	36359	465	
9	2	10	0.5	3940	277	Changing the radius
10	2	20	0.5	3040	233	
11	2	50	0.5	2093	228	
z12	2	100	0.5	1532	192	
13	2	200	0.5	1017	161	
14	2	500	0.5	557	171	
15	2	1000	0.5	299	174	
16	2	2500	0.5	87	121	
17	3	5	0.5	5356	201	Changing the visit count
18	4	5	0.5	5421	136	
19	5	5	0.5	5452	105	
20	7	5	0.5	5476	81	
21	10	5	0.5	5495	62	

Table 1: Different parameter settings show the influence to the resulting places

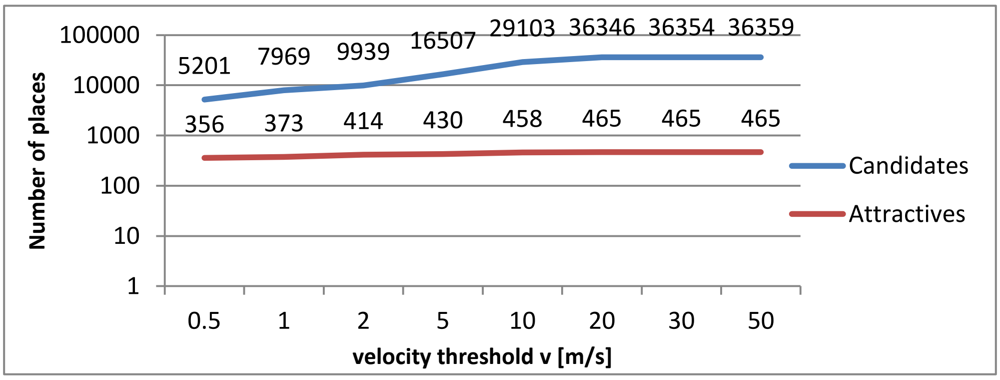


Figure 5: Varying the velocity threshold parameter

Increasing the velocity threshold v (experiments 1-8) leads to an increase in the number of places as then also locations, when paths only geometrically cross without the individual stopping, are being considered. The criteria that determine an individual stays at a place are more often fulfilled. While the upper bound for the number of candidates is equal to the number of tracking points in the dataset, the upper bound for attractive places is calculated by $N_{AttractivePlaces} = N_{TrackingPoints}/n$. Depending on the scenario and the quality of data v has to be adjusted (compare to example 1 and 2 of next chapter).

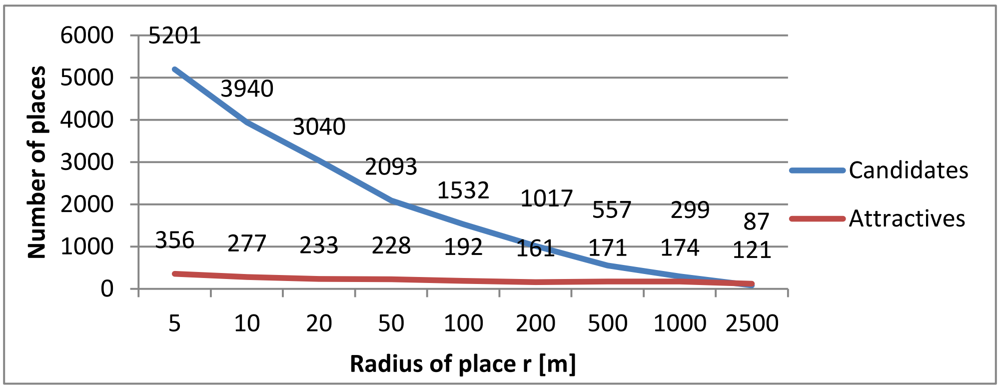


Figure 6: Varying the radius of a place

Varying the size of a place by increasing the radius r (experiments 9-16), decreases the number of candidate and attractive places. The larger the radius becomes the more movements fall into an already existing place and the less candidates are created. Small and adjacent places coincide and are treated as one, especially in areas where the concentration is high. Similar to the velocity threshold, r has to be adjusted to the data as well.

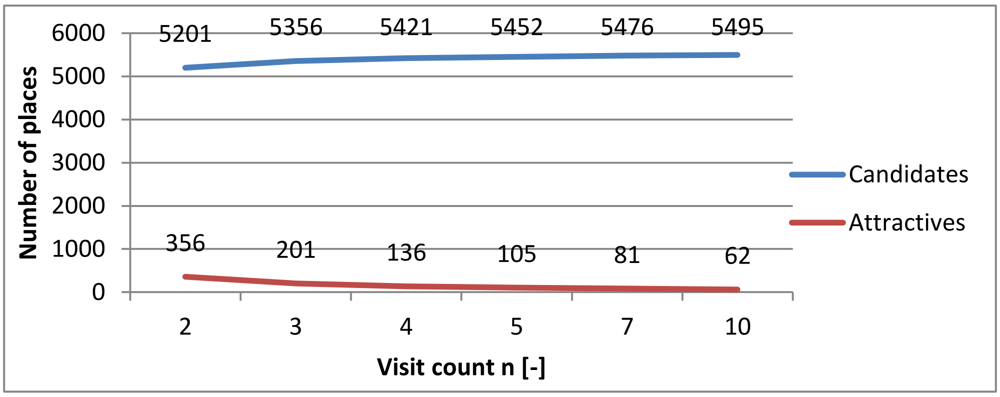
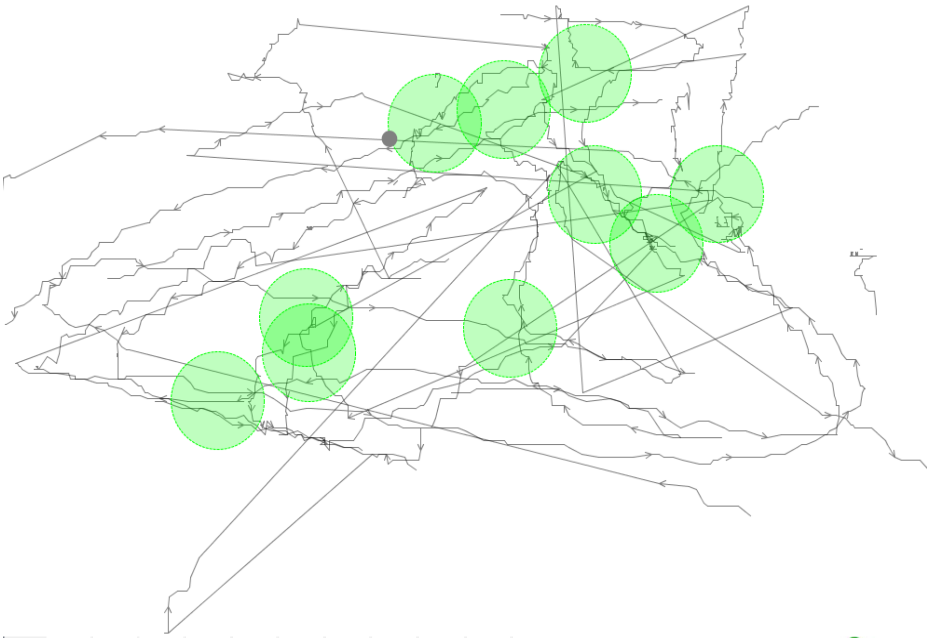


Figure 7: Varying the required visit count parameter

Increasing the count parameter n decreases the number of attractive places (experiments 17-21). If the density of data is low, the decrease will be higher, because the probability that several individuals visit a faraway place is quite low. This parameter has to be adjusted to the density of the dataset.

The previous result and parameter studies refer to the places-extraction step of the algorithm. The following concerns the second and third steps and shows what the results after those steps may look like. The segmentation results can easily be visualized by a graph structure. To this end, we choose another example and present the corresponding segmentation matrix and graph (cf. Figure 8).



	7	8	17	21	27	28	19	45	20	18
7	1	0	1	0	0	0	2	0	0	1
8	1	2	2	0	0	0	0	0	0	1
17	0	1	0	1	0	1	1	0	0	2
21	0	0	0	0	2	0	0	1	0	0
27	0	2	0	0	0	1	0	0	0	0
28	0	0	1	0	1	0	0	0	0	1
19	2	0	1	0	0	0	0	0	2	0
45	0	0	0	1	1	1	0	0	0	0
20	0	0	0	0	0	0	1	0	2	0
18	0	1	2	1	0	0	0	0	0	0

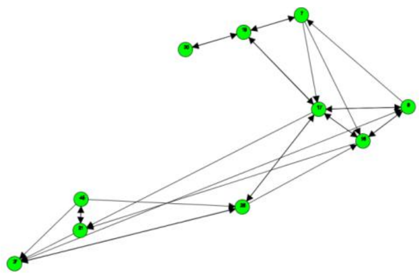


Figure 8: Example for the results after the segmentation step

Figure 9 shows simple examples for the last step of the algorithm, with and without an evaluation of segments of several clusters. There, clustering is applied using the Hausdorff-distances between each segment. The resulting clusters are symbolized by the colors red and blue. In this case some individuals (in red) have used a quite different way to traverse from one to another place. Due to this fact and this small given context, we can consider this movement to be a special one. This may change when there will be more segments within this cluster after a longer observation time.

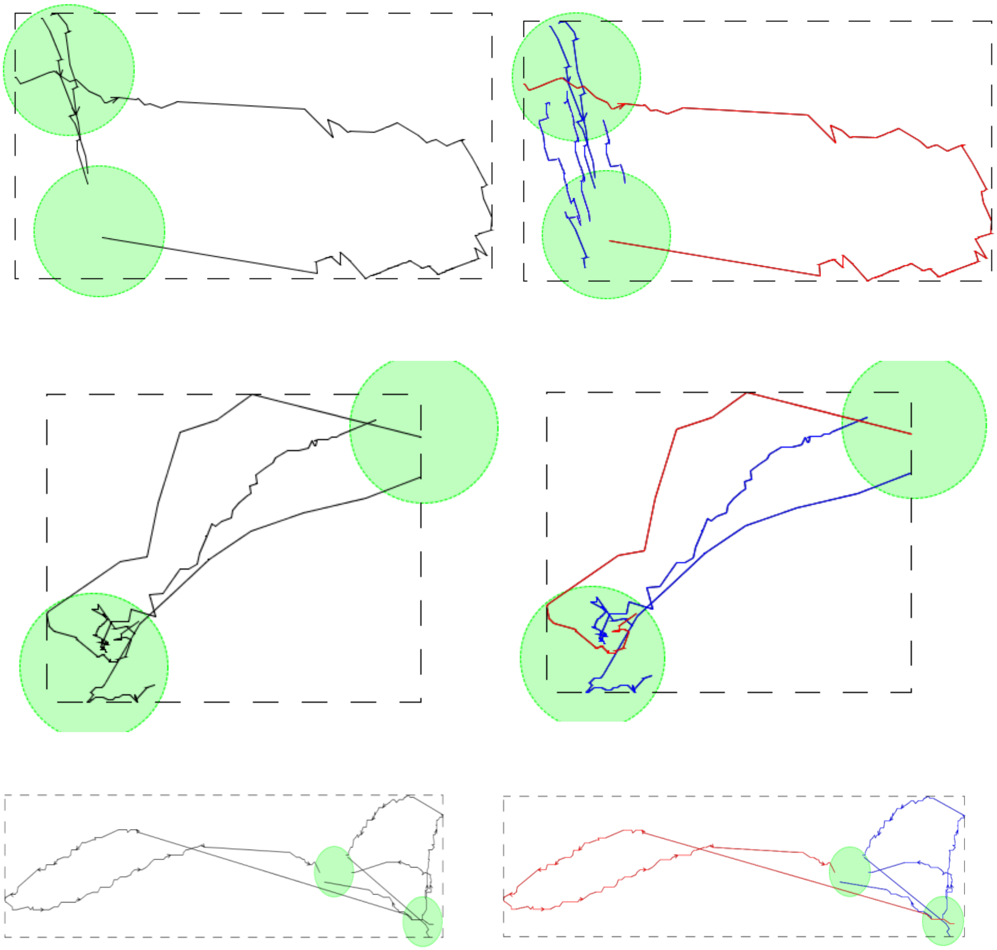


Figure 9: Examples for evaluation step with (right) and without (left) recognition of unusual movements by clustering the trajectory segments

5. Transferability to Other Scenarios

The presented algorithm can be applied to trajectory data provided by different sources like GPS or video tracking. Therefore, we are going to show further examples of the examination of various data types and scenarios. While the first two examples are based on GPS data the last example uses data recorded by video cameras.

In the first example the data (212 trajectories) have been collected by employees of the Institute of Cartography and Geoinformatics while traveling in Hannover, Germany, during a time period of approximately two month. Using the parameters n=2, r=15m and v=0.3m/s leads to 13 attractive places found. Those can be visually inspected and assigned to existing, semantically meaningful places. For this purpose an extract from Google Maps is presented, where the attractive places are marked. Two of them represent tram stops (C, D), one is a crosswalk with traffic lights (B) and another one is the building the institute is located in (A).

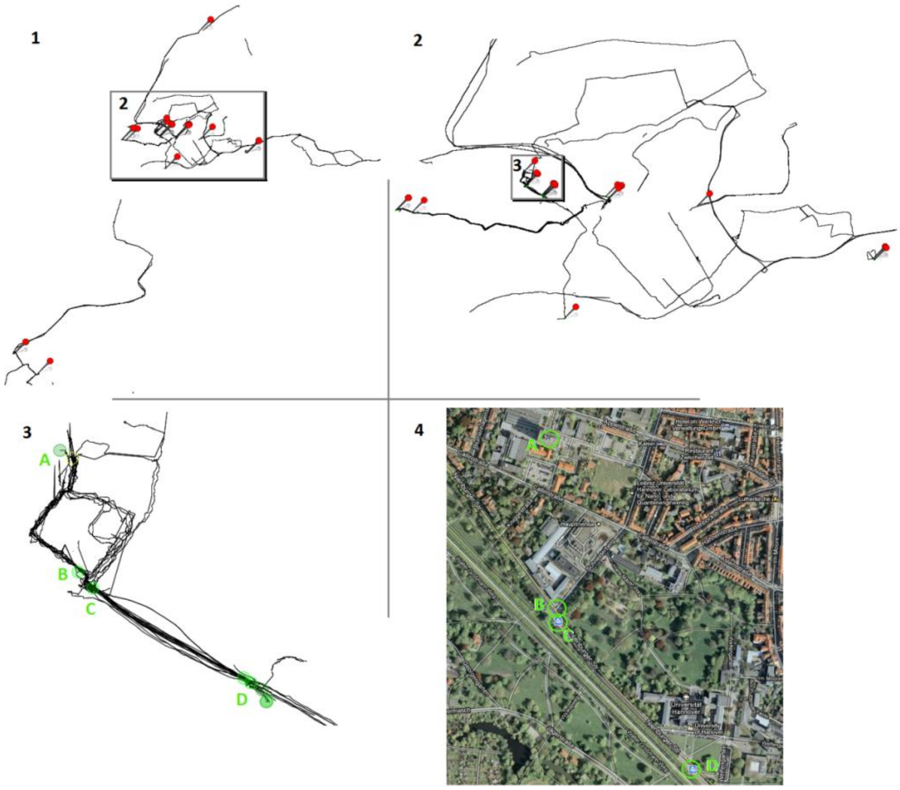


Figure 10: Extraction of interesting places in GPS data set presented in different scales (1-3). An extract of Google Maps for assigning the found places to existing ones (4)

A GPS-game, in which several groups of students participated, provides the data for another example. The results shown in Figure 11 are achieved by using n=2, r=15m and v=0.1m/s as input parameters of the algorithm. Although this example also contains GPS-data, the velocity threshold can be set lower than in the example before, because it is priory known that the students were walking. The starting point of this game was in the left center. Higher densities of trajectories and of interesting places can be recognized there. Most of the places represent either meetings of different groups or road junctions, where the participants stayed for a certain time to plan where to go next.

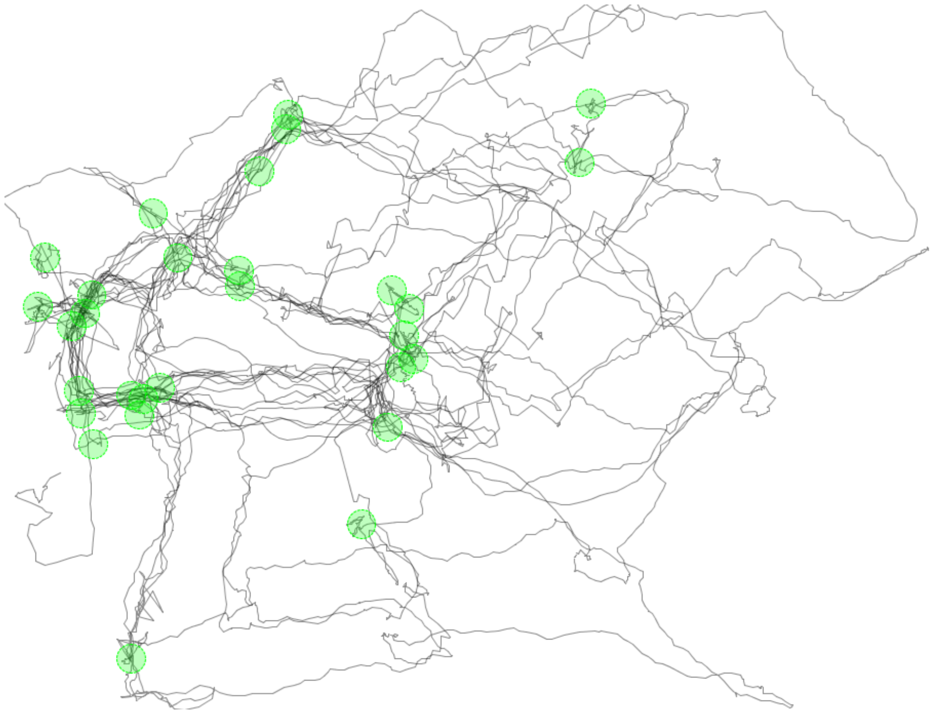


Figure 11: Interesting places found in a data set provided by a GPS-game

Another example originates from a video tracked handball match. One team has been tracked over a period of ten minutes, leading to 7 trajectories with 104993 sample points. This time the following parameters are used: visit count: 3, region radius: 0.5m, velocity threshold: 0.1m/s. Figure 12 (left) shows a snapshot of the court, the seven players of the tracked team (gray dots) and the 14 identified places (green dots) at a certain time. Considering the facts that the tracked team defends on the left and uses a specific defense formation, which is strongly kept by the players, the found places are reasonable. Those places can be interpreted and explained by visual inspection. The places 1 and 2 are places the goal keeper often stands at. The places 3 to 9 can be assigned to positions of a typical defense formation (cf. Figure 12, right). At the place in the center of the court (12) the throw-offs take place. Places 13 and 14 are offensive positions of the left wingman, at places 10 and 11 the right wingman has to wait for the throw off before entering the opposite half of the court.

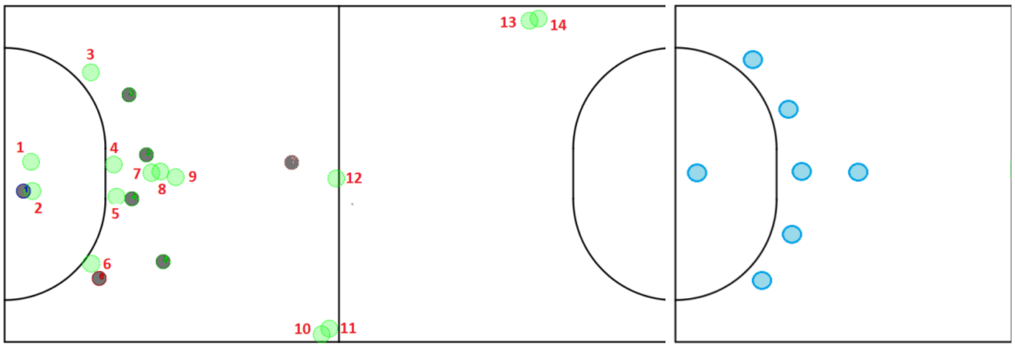


Figure 12: Examination of trajectory data provided by a video tracked handball match. Left: one snapshot with overlaid found interesting places, right: one typical defence formation during a handball match

6. Summary and Outlook

In this paper, we presented a first approach to detect abnormal movements of individuals depending on their environment. The results showed that the method we are using can be the first step to reach our overall goal. After finding typical behavior, we will be able to determine the deviations thereof, which we consider as abnormal.

This approach minimizes the data volume and computational costs by generating spatial models of movement behavior and incrementally updating with observed movements. The update mechanism does not require any reprocessing of data from previous observations. The unique processing, which consists of storing the extracted information in a more general model, reduces the amount of data. This way, for computation of a single observation we achieve a favorable runtime complexity based solely on the size of our model, making the algorithm suitable for real-time application. Given the spatial characteristics of the used model, a further speed-up of the used algorithm is possible e.g. by using spatial indexing structures.

As the interpretation of the trajectories is organized in an incremental fashion, it can also be designed in a decentralized way to achieve scalability with respect to the number of objects to track and interpret. This decentralize the algorithms is one of our main topics.

Another topic of ongoing work is to analyze and evaluate the migration graph structure. While doing this, the nodes can be classified by characteristics of entering and exiting trip edges. Next to solving typical tasks like finding the shortest or most favorite paths, many well-known concepts from graph theory can be directly applied to classify parts of the graph. We may identify places that act similar to sources and sinks (many ingoing/outgoing trips, only one outgoing/ingoing trip respectively), hubs (many ingoing and outgoing trips), loops (trips starting and ending at the same place) and so on. The graph structure can also be used for movement prediction. To this end, probabilities of possible target nodes can be calculated by including several factors like relative frequency of edge usage or target distances.

The approach is general enough to be used in several kinds of applications. In the context of LBS or pedestrian navigation, with this method popular places and frequent routes can be identified. Typical routes are also of interest for several planning purposes, e.g. city planning or traffic planning (see e.g. van der Spek, et al. (2009)).

Since in some parts of our method values have to be set a priori, it is not able to handle different scenarios automatically. It also does not adjust to varying situations. Therefore, an auto-fitting or learning technique to determine the three parameters would be very helpful.

Besides further things we are planning to deal with, the consideration of the temporal domain is important, e.g. to distinguish between trips common at typical times of day (see Makris and Ellis (2002)).

7. References

Andrienko, G., Andrienko, N., Bak, P., Keim, D., Kisilevich, S. and Wrobel, S., 2011. A conceptual framework and taxonomy of techniques for analyzing movement. Journal of Visual Languages & Computing 22(3), 213–232.

Ashbrook, D. and Starner, T., 2003. Using GPS to learn significant locations and predict movement across multiple users, Personal & Ubiquitous Computing 7: 275-286.

Baiget, P., Sommerlade, E., Reid, I., and Gonzàlez, J., 2008. Finding prototypes to estimate trajectory development in outdoor scenarios. In Proceedings of the First International Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (THEMIS2008), September

Buchin, K., Buchin, M., van Kreveld, M., and Luo, J., 2009. Finding long and similar parts of trajectories. In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '09). ACM, New York, NY, USA, 296-305.

Dodge, S., Weibel, R., and Lautenschütz, A. K., 2008. Towards a taxonomy of movement patterns. Information Visualization, 7, 240–252.

Kang, J.H., Welbourne, W., Stewart, B. and Borriello, G., 2005. Extracting Places from Traces of Locations, ACM Mobile Computing and Communications Review, 9(3).

Krumm J. and Cao L., 2009. From GPS Traces to a Routable Road Map // 17th ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, ACM-GIS 2009, November 4-6, Seattle, Washington, USA, Proceedings. - Seattle

Laube P., Duckham M. and Wolle T. , 2008. Decentralized Movement Pattern Detection amongst Mobile Geosensor Nodes // Geographic Information Science, 5th International Conference, GIScience 2008, Park City, UT, USA, September 23-26, Proceedings. - Heidelberg : Springer

Makris, D. and Ellis, T., 2001. Finding paths in video sequences, in Proc. Brit. Machine Vision Conf., vol. 1, Manchester, U.K., pp. 263–272.

Makris, D. and Ellis, T., 2002. Spatial and probabilistic modeling of pedestrian behavior, in Proc. Brit. Machine Vision Conf., vol. 2, Cardiff, U.K., pp. 557–566.

Van der Spek S., Van Schaick J., De Bois P., De Haan R., 2009. Sensing Human Activity: GPS Tracking. Sensors. 2009; 9(4):3033-3055.