Directional Analysis of Trajectories Based on Trajectory Smoothing

Praveen Kumar Tripathi
University of Texas at Arlington
praveen.tripathi
@mavs.uta.edu

Madhuri Debnath
University of Texas at Arlington
madhuri.debnath
@mavs.uta.edu

Ramez Elmasri
University of Texas at Arlington
elmasri@cse.uta.edu

ABSTRACT

In this article we propose a framework to discover interesting directional patterns in trajectory data sets. The proposed framework has five stages; trajectory smoothing, directional segmentation, directional classification, filtering and finally clustering. The main contributions are in the stages for smoothing, directional classification and filtering. Trajectory smoothing is an important step in the analysis of complex, non-smooth trajectories data sets, such as animal movement data. In directional classification stage, different sub-trajectories are assigned to the classes corresponding to their directional orientation. In the filtration stage the outlier trajectories are removed from the respective classes using a novel convex hull based approach. We used animal movement data in this work.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data mining

Keywords

Trajectory, clustering, spatio-temporal

1. INTRODUCTION

Trajectory data is an example of spatio temporal data where the spatial location as well as the time order associated with each data point is very important. Some examples of spatio temporal data are tracking data in applications like cell phone and vehicle tracking, hurricane and storm tracking data, and animal movement data [4, 5, 10]. The directional aspect of trajectory analysis is very important in various applications [11], for example in map matching [3] and in direction based query processing [4]. This kind of analysis is very useful in applications such as analyzing weather data (hurricane tracks), public transport data (GPS) and animal movement data.

The first stage of the proposed framework deals with *trajectory smoothing*, which is important in analyzing trajectories

MELT'15 , November 03-06, 2015, Bellevue, WA, USA

DOI: http://dx.doi.org/10.1145/2830571.2830771

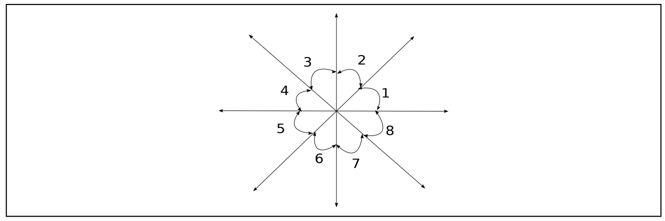

Figure 1: Small angles (45°)

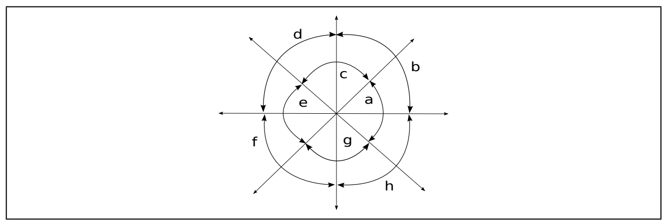

Figure 2: Large angles (90°)

that exhibit non-smooth characterstics. Trajectory smoothing (simplification) has been addressed in [6, 11]. Non-smoothness is marked by frequent sharp directional changes. The trajectories in animal movement data exhibits this non-smooth property. In the smoothing step we approximate the original trajectories by eliminating these sharp angular turns to focus on directional characteristics of the trajectory.

In the next stage we do *directional segmentation* of the trajectories. We impose directional consistency on the trajectories (sub trajectories) by allowing them to deviate initially by a maximum of 45°, and eventually by a maximum of 90°. If any trajectory shows a deviation more than 90°, we split it into sub-trajectories such that the directional consistency is maintained. We consider 16 directional ranges shown in Figure 1 and Figure 2. Each sub-trajectory is assigned to one of these 16 classes. Analysis of trajectory data after segmentation has been a well studied domain in spatio temporal data, for example [5, 8–10].

The filtration stage removes outlier sub-trajectories from the directional classes to focus on important directional ranges. We considered two approaches for this task. The first approach uses the *minimum bounding rectangle (MBR)*, whereas the second one uses a novel *convex hull (CH)* based approach. The MBR based approach results in big spaces around a trajectory; hence it is a very vague approximation of the trajectory. The CH based approach approximates the trajectories to a much closer extent. We finally use a modified DBSCAN [7] algorithm to identify the inherent clusters, which capture the significant directional patterns in the data sets. We used animal movement data [1] that consists of movement of Elk in 1993 and comprises 33 trajectories and 20065 data points.

2. RELATED WORK

The article [9] proposes a partition-and-group framework for clustering trajectory data. Using the concept of minimum descriptive length (MDL) the original trajectories are segmented. These segments are called trajectory partitions, which are then clustered using a modified version of DBSCAN algorithm, which clusters the line segments. Finally the clusters are represented by representative trajectories.

In [8], a clustering algorithm is given for the trajectory data that uses a combination of techniques from data mining, computational geometry and string processing. Trajectories are pre-processed followed by segmentation and classification. The next phase finds the frequent occurring sub-strings; these are called $motifs$, and then maps the sub-trajectories corresponding to the motifs to some feature space. The next stage performs density based clustering and the final stage does the post processing of the clusters.

In [10], the trajectory clustering technique of [9] has been extended for trajectory classification. Two levels of clustering, namely region-based and trajectory based clustering, are done. Clustering is used to find the discriminative features for classification.

Trajectory smoothing has been addressed in [6, 11]. In [11] authors propose a trajectory smoothing method that preserves the direction information. The aforementioned works are similar to the present work on the basis of trajectory segmentation followed by the clustering of the trajectory segments. However, our contribution is on the directional consistency, so our trajectory segmentation method is purely direction based. The other contributions are trajectory simplification and convex hull based outlier removal concepts.

3. PROPOSED FRAMEWORK

Before we describe the proposed framework, we define a trajectory and its major defining characteristics. Let us consider, a trajectory data set D comprising of n trajectories viz., $D = (Tr_1, Tr_2, \ldots, Tr_n)$.

Trajectory Definition: A trajectory Tr_j of size s is a sequence of points $[p_1, p_2, \ldots, p_s]$, where p_1 is it's initial point and p_s is the final point. An i^{th} point p_i in Tr_j is associated with spatial co-ordinates (x_i, y_i) and the associated time t_i.

Trajectory Segments (L_k): A trajectory Tr_j consists of line segments $L_k = \overline{p_k p_{k+1}}$ which are formed by joining the k^{th} and $(k+1)^{th}$ consecutive points in it, where $k \in [1, \ldots, s-1]$.

Angular Attribute (θ): This is a very important attribute of a trajectory for directional analysis, which considers the angles between its L_k and L_{k+1} consecutive line segments.

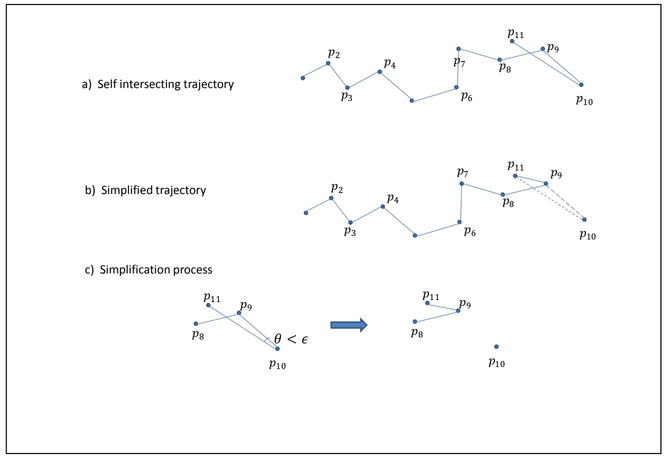

Figure 3: Self intersecting Trajectories

This involves three successive points: p_k, p_{k+1} and p_{k+2}.

$$\theta = min(\angle(\overline{p_k p_{k+1}}, \overline{p_{k+1} p_{k+2}}), 360^\circ - \angle(\overline{p_k p_{k+1}}, \overline{p_{k+1} p_{k+2}})) \quad (1)$$

where, angle between the two line segments $\overline{p_k p_{k+1}}$ and $\overline{p_{k+1} p_{k+2}}$ is $\angle(\overline{p_k p_{k+1}}, \overline{p_{k+1} p_{k+2}})$. Angle is measured in anti-clockwise rotation from $\overline{p_k p_{k+1}}$ to $\overline{p_{k+1} p_{k+2}}$. We consider the smaller of the angles between the two line segments so that $0 \leq \theta \leq 180^\circ$. A large value of θ represents a small change in direction whereas a small value represents a sharpe change. $\angle$ is the symbol for absolute value of an angle.

The proposed framework consists of the following stages: *1) trajectory smoothing*, *2) trajectory segmentation*, *3) directional classification*, *4) convex hull based sub-trajectory filtration*, and *5) clustering*.

3.1 Trajectory Smoothing

The animal movement trajectories [1] tend to be very haphazard and non-unifromly sampled. For meaningful trajectory data analysis, trajectory smoothing becomes imperative. Our smoothing of trajectories considers self-intersecting trajectories (see Figure 3), which are not explicitly addressed in previous work.

Trajectory Smoothness ($sm\,(Tr)$): Smoothness of a trajectory determines its directional consistency over its whole length. As a smoothness measure, we consider the mean ($sm_\mu\,(Tr)$) as well as the standard deviation ($sm_{sd}\,(Tr)$) of the anglular attributes of a trajectory. If $\overline{\Theta}$ is the vector of the angular attributes of a trajectory Tr, then : $sm_\mu(Tr) = mean(\overline{\Theta})$, $sm_{sd}(Tr) = sd(\overline{\Theta})$. A large $sm_\mu(Tr)$ and a small $sm_{sd}(Tr)$ would indicate a smooth trajectory, whereas small values for $sm_\mu(Tr)$ indicate a very jagged trajectory. The simplified (approximated) trajectory should have better smoothness to highlight its main directional properties. If an initial trajectory is Tr, then the goal of the smoothing process is to simplify it to a trajectory Tr_s such that $sm_\mu(Tr) \leq sm_\mu(Tr_s)$ and $sm_{sd}(Tr) \geq sm_{sd}(Tr_s)$.

To address the unevenly sampled trajectories, we segment a bigger trajectory into sub-trajectories at a point where, the time difference between the successive sampled points is larger than a given threshold (30 minutes for animals).

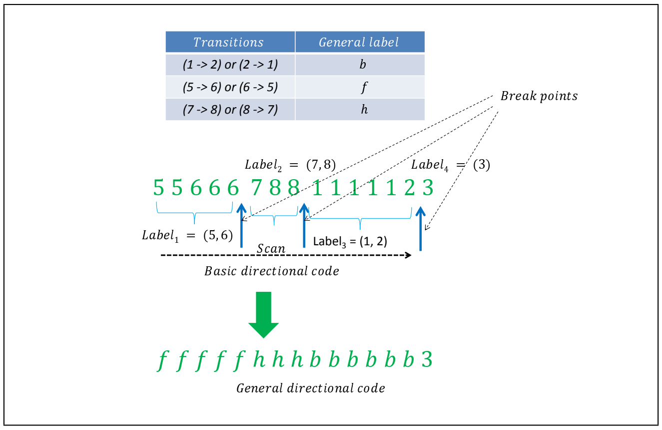

Figure 5: Basic to General direction encoding

Figure 3 shows how we deal with the self-intersecting trajectories as part of the smoothing process. The segment $\overline{p_{10}p_{11}}$ intersects the segment $\overline{p_8p_9}$ because the angle between the segment $\overline{p_9p_{10}}$ and $\overline{p_{10}p_{11}}$ is very small. As mentioned earlier, small angles indicate sharp turns leading to non-smooth trajectories. Self-intersecting trajectories are the results of extreme cases of small angles. Using small angle threshold ϵ, if an intermediate angle is less than this threshold, we filter it out by discarding the intermediate point (p_{10} in this case). This will result in the formation of the new segment $\overline{p_9p_{11}}$. If the resulting angle is still less than ϵ, further smoothing is applied.

3.2 Trajectory Segmentation

In this step of the proposed framework, trajectories are segmented into sub-trajectories belonging to one of the 16 directions depicted in Figure 1 and Figure 2. Eight of these directions (Figure 1), are 45° apart covering the whole 360° angular space, whereas the remaining ones (Figure 2) are 90° apart. This choice of the smaller angle directional segment viz., 45° and a larger overlapping directional segments viz., 90° was made so that a trajectory that moves along a boundary between two 45° regions, does not keep being segmented into very short segments. For example, if a trajectory moves between region 1 and region 2, a longer sub-trajectory in region b would be created.

Initially we examine the slope of the line segments forming the trajectory, and label the line segments according to their slope. Since the basic angle for segmentation in this work is 45°, the total number of these labels is $360/45 = 8$. Therefore the label for a particular line segment will be the quotient when the angle of the slope is divided by 45. The example in Figure 4 corresponds to a trajectory from hurricane data. Using the basic directional encoding algorithm, the trajectory is now labeled as (556667881111123) (Figure 4).

The next step works on the basic directional encoding obtained and scans the basic directional code of the trajectory from left to right and looks for a break point (please see Figure 5). The break point is the position along the basic directional encoding where, the direction changes by 2 or more labels. Along with the break point a *Label* list is also stored, this list will hold the basic angular transitions so that it may be accordingly encoded in the general directional code. For example in Figure 5, break points create 4 sub-trajectories 55666, 788, 111112 and 3, the last is discarded as too short. Corresponding $Label_i$ lists have the entries $(5, 6)$, $(7, 8)$ and $(1, 2)$, respectively. These transitions correspond to the general angle code of f, h and b respectively (Figure 2, Figure 5). The break point will act as a marker while segmenting the original trajectory into its directional sub-trajectory. Here the output will be ($fffffhhhbbbbbb$). The final segmentation returns the three sub-trajectories which are ($fffff$), (hhh) and ($bbbbbb$).

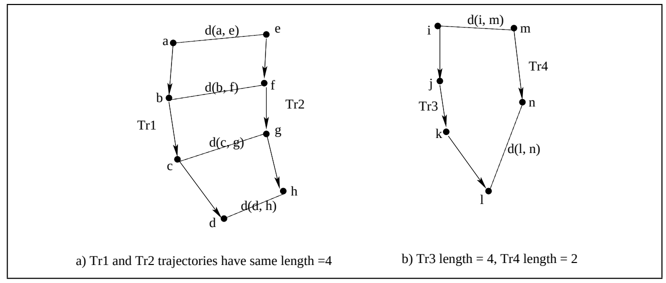

Figure 6: Distance between two sub-trajectories

We use convex hull (CH) to approximate each trajectory, and use convex hull intersecting theorem called *separating axis theorem* to filter the outliers [2].

3.3 Clustering

After removing the outlier sub-trajectories using the CH based technique, we do the clustering of the remaining sub-trajectories to obtain the final directional patterns. We use a modified version of DBSCAN [7], which is a density based clustering algorithm. This is very similar to the one used in [9], except for the fact that here we have the sub-trajectories as the basic data to be clustered instead of line segments. We omit the details of DBSCAN due to space limitations, see [7, 9] for details.

For the DBSCAN algorithm we need a distance measure to find the similarity between the sub-trajectories. Figure 6 shows how we computed the distance between the two sub-trajectories. There are two possibilities, Figure 6(a) where the two sub-trajectories have the same length, and Figure 6(b) where the length of the two sub-trajectories is not the same. For the first case we simply consider the average of the distance between the sequential points in the sub-trajectories. In the second case we consider only the average of the distance between the first points and the last points in the sub-trajectories. In Figure 6(a) for example the distance will be:
$D(Tr_1, Tr_2) = (d(a, e) + d(b, f) + d(c, g) + d(d, h))/4$ whereas, in Figure 6(b) it will be:
$D(Tr_3, Tr_4) = (d(i, m) + d(l, n))/2$. In this atricle the distance $d(x, y)$ is the $Haversine\ formula$.

4. EXPERIMENTS AND RESULTS

The framework has been implemented in MATLAB. We use animal data set as described in Section 1. The impact of smoothing is evident in Table 1, as sm_μ has been improved (is higher) and, the value of sm_{sd} is lower after the application of smoothing, compared to that of the original trajec-

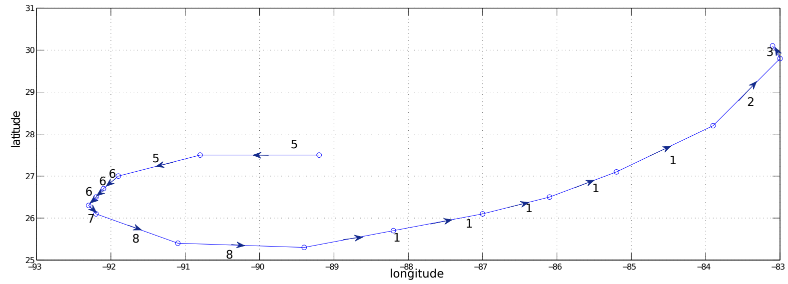

Figure 4: Basic directional encoding

Table 1: Effect of Smoothing on Animal (Elk) data set

$Trajectories$	$sm_{\mu}(Tr)$	$sm_{sd}(Tr)$
Original Animal Data	75.93	55.72
Smoothed Animal Data	87.96	51.08

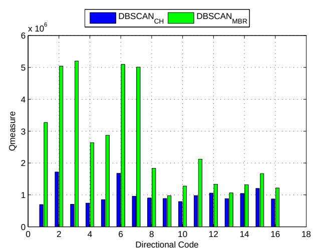

Figure 7: Animal Qmeasure

tories. These results are the aggregate values considering all the trajectories together.

For clustering (DBSCAN algorithm), we use $Min_{line} = 7$ (minimum number of sub-trajectories in the neighborhood of a trajectory for density consideration) and Min_{dst} (neighborhood radius) has been computed as per the suggestions in [9].

Qualitative analysis of clusters: We used the quality measure proposed in [9]. This measure is given below:

$$\sum_{i=1}^{num_{clus}} \left(\frac{1}{2C_i} \sum_{x \in C_i} \sum_{y \in C_i} dist(x,y)^2 \right) + \frac{1}{2|N|} \sum_{w \in N} \sum_{z \in N} dist(w,z)^2 \quad (2)$$

where, num_{clus} is the number of clusters, N is the set of noise sub-trajectories and C_i is the i^{th} cluster. This quality measure computes the sum of the square error (SSE), which means the smaller this value, the better will be the clustering result.

In order to show the effectiveness of the proposed CH based filtration method over the MBR based method we present the results in Figure 7 using the evaluation method given in Equation 2. The clustering results using CH as the filtration approach ($DBSCAN_{CH}$) has resulted in lower values of $Qmeasure$ compared to the MBR based method ($DBSCAN_{MBR}$) when applied to animal movement data for all the directional classes.

5. CONCLUSION

In this article we proposed a novel framework for the directional analysis of trajectory data. We proposed a new trajectory smoothing approach as well as a novel CH based filtration method using convex hulls of subtrajectories. A novel technique for identifying the directional orientation of the trajectory data was proposed. This framework will be very useful in finding the directional movement patterns of animal trajectory for studying their behaviour and their habitat.

6. REFERENCES

[1] http://www.fs.fed.us/pnw/starkey/data/tables/.

[2] https://en.wikipedia.org/wiki/Hyperplane_separation_theorem.

[3] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. Proc. VLDB '05, pages 853–864, 2005.

[4] S. Brakatsoulas, D. Pfoser, and N. Tryfona. Modeling, storing, and mining moving object databases. In *Proc. IDEAS '04*, pages 68–77, 2004.

[5] M. Debnath, P. K. Tripathi, and R. Elmasri. A novel approach to trajectory analysis using string matching and clustering. In *ICDM Workshops*, pages 986–993, 2013.

[6] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973.

[7] M. Ester, H. P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.

[8] J. Gudmundsson, A. Thom, and J. Vahrenhold. Of motifs and goals: mining trajectory data. In *Proc. SIGSPATIAL '12*, pages 129–138, 2012.

[9] J. G. Lee and J. Han. Trajectory clustering: A partition-and-group framework. In *In SIGMOD*, pages 593–604, 2007.

[10] J. G. Lee, J. Han, X. Li, and H. Gonzalez. Traclass: Trajectory classification using hierarchical region-based and trajectory-based clustering. *Proc. VLDB Endow.*, pages 1081–1094, August 2008.

[11] C. Long, R. C. W. Wong, and H. V. Jagadish. Direction-preserving trajectory simplification. *Proc. VLDB Endow.*, pages 949–960, August 2013.