Mining Dense Trajectory Pattern Regions of Various Temporal Tightness

Ms. Sumaiya I. Shaikh[1], Prof. K. N. Shedge[2]

[1] *Ms.Sumaiya I. Shaikh, ComputerEngineering Department,SVIT, Chincholi, Nashik, Maharashtra, India*
[2] *Prof. K. N. Shedge, ComputerEngineering Department,SVIT, Chincholi, Nashik, Maharashtra, India*

ABSTRACT

Mining UT patterns is the process of discovering relationship between moving objects. Traditional methods of UT pattern mining can only detect individual trajectory also end user does not have an idea about which type of trajectories are hidden into input dataset. There are two types of phases involved in proposed UT mining framework such as, initial phase and granularity phase. In first phase, initial patterns get discovered. By implementing TraClas method sub-trajectory clusters are formed. In second phase detail level of each pattern get extracted. With the detail information of trajectories pattern forest is constructed. It contains different types of trajectories in it. The proposed system contributes dense trajectory representation which guarantees a compact coverage of foreground motion as well as of the surrounding region. Experimental results demonstrate that our framework facilitates easy discovery of various patterns from real-world trajectory data.

Keyword : - *Trajectory pattern mining, synchronous movement patterns, moving object trajectories, trajectory clustering*

---

1. Introducation

Recent development on satellite, sensor, RFID, video, and wireless technologies made possible to track movements of objects and gathered large amounts of trajectory data. Trajectory mining has many applications such as, animal movement data, ship navigation data, and person tracking data etc. While interacting and communicating with each other objects shows synchronous movement patterns. Synchronous movements of patterns such as, a group of moving objects move together, group of objects race another set of moving objects with a small time delay. Moving objects also represents the asynchronous movement patterns such as, group of moving objects follows same path every year. Asynchronous patterns also known as, unifying trajectory pattern. Generally, unifying patterns can be defined as, the group of moving objects closely related with respect to time and location. UT patterns can be observed in deer migration as they move together at the same time hence they are always close to each other. Unified trajectory mining is useful to learn interactions between moving objects and possibly group dynamics. Applications like, zoology, sociology, etc, UT patterns are widely used. Previously, many efforts have been dedicated to discovery of trajectory patterns. Previous studies have been conducted to identify flock patterns, convoy patterns, swarm patterns, moving patterns etc. Each of this is identical to one type of trajectory pattern. Therefore, UT patterns mining became tedious and inefficient task. Determining temporal rigidity is the nice way for classification of trajectory patterns. This is the motivation of our proposed unifying trajectory pattern mining. In this work, after trajectory dataset is loaded in the system or given as input to the system, initial cluster will be created using TrakClas method. In the process of clustering or sub-trajectory clustering, trajectories are divided into group of line segments. As per analysis of these trajectories similar trajectories according to the spatial similarities are then grouped together into a cluster. Each cluster contains trajectories which are similar and closed to each other in terms of location and time. Projection point is determined using rotations metrics. End point of trajectory is estimated onto average direction [8]. Next step is to identify set of reference movement of critical patterns of partitioned trajectories. In the process identifying a set of reference movements, as per similarity measure each trajectory partition is assigned to the closest reference movement. Aim behind this procedure is to increase the compression of trajectory partitions.

There are two main advantages of proposed system is user does not have to specify any input parameter or in processing phase there is no need of user interference. Another is it return the sub-trajectories by maximizing the ratio of data compression. For maximization of data compression two properties are retain by UT-patterns such as, preciseness and conciseness. MDL i.e. Minimum Length Principle has been proposed in this work, it is used to describe useful model classes, to describe brief algorithmic description. If split is decreases the MDL cost then drill-down approach is utilized for the purpose of deriving multiple time-constrained or time relaxed constrained. Drill-down approach minimizes the cost of MDL principle. Reverse to the drill-down approach Roll-up approach does not decrease the cost of MDL. These both approaches are the part of OLAP-operations. Both are useful in the process of pattern forest construction.

As a part of contribution proposed system also identifies the dense area.

2. RELATED WORK

G. Lee, J. Han, et al., discussed about trajectory patterns that arranged according to the strength of temporal constraints. The proposed framework in this paper consists of two phases: first is initial pattern discovery and the second is granularity adjustment. In the initial phase detail levels of patterns are discovered. In the other phase patterns are merge together to construct a forest. In this paper, UT-pattern mining algorithm is developed. The algorithm first discovers initial UT-patterns using the intuitive information-theoretic principle of maximizing data compression and then constructs a pattern forest by drill-down and roll-up to discover more patterns. Finally, UT patterns are compared with the flock patterns. Flock patterns are classified as time-constrained. In this author like to claim flock patterns that are sometime too restrictive to find useful pattern. In this paper, use synthetic data sets created by varying four control parameters. Author discussed about, flock patterns, time-relaxed trajectories, sub trajectory cluster etc [1].

I P. Laube and S. Imfeld, proposed a techniques for spatio-temporal analysis. It is relative motion within set of moving point objects, for e.g. GPS-tracked animals. In their research work they aim to construct flexible analysis concept for the integrated analysis of motion parameters of groups of moving point objects. They proposed REMO model to identify interrelationships among in any kind of observation data of moving point's objects. They defined pattern identification is the process of associative motion within groups of moving point objects also to identify sub-groups according to equal or similar movements. Discovery of patterns over time means identifying the concerned individuals and their location and extent on the time axis [2].

P. Laube, M.V. Kreveld et al. suggested REMO model which investigates object's motion properties in an analytical space as well as spatial constraints of the object's lifelines in geographic area. In their proposed research work they represented some geometric assets of the denominated patterns with respect to their efficient computation [3].

In [4], they did the study of development of a generic approach. It is to discover geographic knowledge (GKD) in partial lifeline data. It contains some necessary steps like, data reduction and projection, exploratory analysis and model selection, Visualization etc.

In[5] M.Benkert, et al, research was conducted on reporting of floack patterns. In this research they were analysed that tree-based algorithm is suitable for discoverying flock patterns. But it depends on the characteristics of input set [3]. Similarly, stratergy is implemented by M. Nanni and D. Pedreschi in [13]. To mine trajectories of moving objects they were defined a time-focused clustering. Temporal focussing, is sketched which is new approach to the trajectory clustering problem. It aims to exploit the intrinsic semantics of the temporal dimension for improvement of trajectory clustering quality. They implemented a density-based clustering method is utilised for moving objects trajectories.

P. Bakalov, et al [6] suggested basic symbolic join algorithms about time relaxed trajectory joins manifested on.

Traditionally, there was two kind of approaches are represented in this research, from both of this first approach is based on notion of multiple origins and the other is hueristic solution based on "spilt and merge" method[6]. This approaches are suitable where there is limited memory resources. Longest duration flock pattern computation problem is discussed in[9],.

D. Sacharidis et al. [7], determines hot motion path which is also known as, time relaxed trajectory joins. It detects frequently traveled trails of numrous moving objects. A distributed settings is considered,due to it's co-ordinators maintains the hotness & geometrics of this paths. The proposed work is limited to freely moving objects. A framwork based on partitioning and grouping stratergy is also called as, Sub-trajectory clusters. It is implemented for trajectory clustering. TRACLUS method is used to construct sub-trajectory clusters. Main aim of TRACLUS algorithm is to disovered sub-trajectories from huge trajectory dataset. J. Gil Lee, et al[8], represented trajectory clustering algorithms. It is used to club trajectories having similar type or attributes. Their main focused is to detect common sub-trajectories. In proposed research work they introduced a new partition-and-group framework for clustering trajectories, which partitions a trajectory into a set of line segments, and then, groups similar line segments together into a cluster. The main benifit of proposed framework is to identify basic sub-trajectories from database of trajectories.

T. Brinkh[10], represented the framework to evaluate spatio-based temporal database for generating network-based moving objects as many applications dealing with the spatio temporal data. sub-trajectory clustering utilizes hueristic solution based on divide and conquer method. Clustering moving objects is an interesting approach to catch regularities of the moving objects.

Y. Li, et al. [11], demonstrated clustering analysis on moving objects. It provides some interesting pattern changes. The concept of micro-cluster is introduced by them to detect some regularity of moving objects and handles large databases. An efficient algorithm is implemented to keep moving micro-clusters graphically small.

C. Bohm, C. Faloutsos, et.al [12], proposed a robust framework for determining a natural clustering of a given data set, based on the minimum description length (MDL) principle. The proposed framework, Robust Information-theoretic Clustering (RIC), is orthogonal to any known clustering algorithm: given a preliminary clustering, RIC purifies these clusters from noise, and adjusts the clustering's such that it simultaneously determines the most natural amount and shape (subspace) of the clusters. The proposed RIC method can be integrated with any clustering technique ranging from k-means to k-medoids. RIC framework is very flexible, with several desirable properties that previous clustering algorithms don't have. More importantly, the RIC framework does not compete with existing (or future) clustering methods: in fact, it can benefit from them! If a clustering algorithm is good, proposed RIC framework will use its grouping as a starting point, it will try to improve on it and, it will either improve it.

3. PROBLEM STATEMENT

"Mining Dense Trajectory Pattern Regions of Various Temporal Tightness"

4. PROPOSED SYSTEM

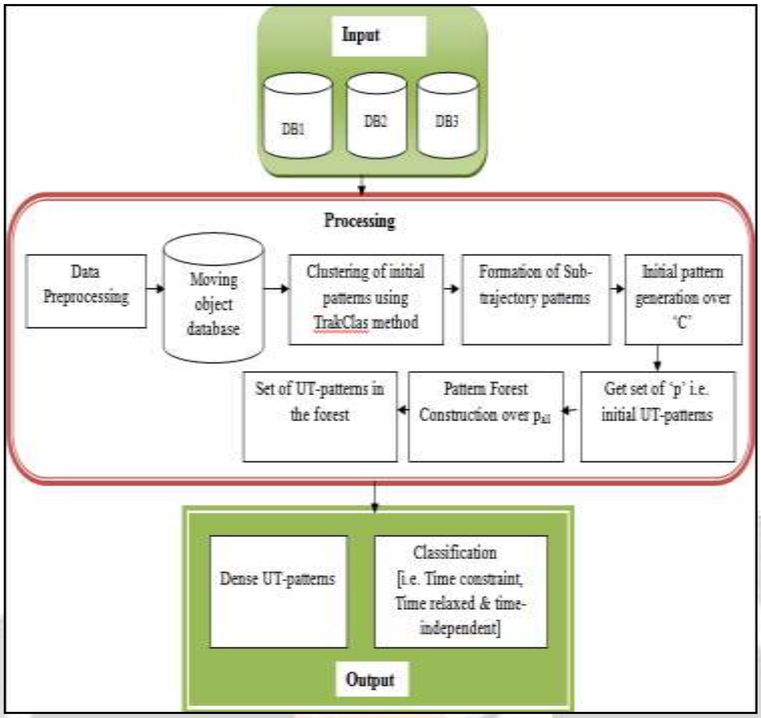

Fig.1 System Architecture

Figure 1 represents the architecture of UT Framework. It contains the large database of various object movements such as, animal data, vehicle data etc. Database is firstly preprocessed and clean and then stored in object movement dataset with spatial and temporal attributes. This database is given as input to the phase I to discover the detail levels of patterns. The output is then passed to Phase II to adjust the different levels of patterns. It is further used for classification according to time restricted patterns, time delayed patterns etc. And after successful classification of patterns they are used for various applications ecological analysis, mobility management, traffic analysis, planning and control etc.

5.ALGORITHMS

1. UT-Pattern Mine

Input: A set of Trajectories I= {TR_1, TR_2 …TR_{numtra}}

Output: A set of UT-patterns O= {UT1, UT2 …UT_{numpat}}

Processing Steps:

- Phase I-Initial pattern discovery

2. Perform sub-trajectory clustering over I based on the TRACLUS algorithm[8]
3. Get all sub-trajectory clusters C_{all}
4. For each C ∈ C_{all} do
5. /Algorithm 2 */
6. Execute initial pattern generation over C
7. Get set P of UT-patterns as a result
8. Accumulate P into a set P_{all}
9. End for
10. Phase II-Granularity Adjustment
11. /Algorithm 3 */
12. Execute pattern forest construction over P_{all}
13. Return the set of UT patterns in the forest
14. Classify pattern into three type's i.e. time-constrained pattern, time-relaxed patterns and time independent patterns.

2. Initial Pattern Identification:

Input: A set L of trajectory partitions in a cluster C
Output: A set P of initial UT-patterns
Processing steps:

1. $L1 \leftarrow L, \overrightarrow{R_1} \leftarrow$ DeriveRefMovement(L1);
2. $P \leftarrow \{(\overrightarrow{R_1}, L_1)\}$;
3. Repeat
4. Choose the m^{th} UT-pattern from P, where,
   $M = \underset{(\overrightarrow{R_m}, L_m) \in P}{\operatorname{argmax}} C(\overrightarrow{R_m}, L_m)$;
5. Split the m^{th} UT-pattern into two splits
6. Choose the pair of trajectory partitions, where
   $(Lp,Lq) = \underset{Lp,Lq \in Lm}{\operatorname{argmax}} dist(Lp,Lq)$;
7. Distribute t-partitions of Lm into two
8. $L^p_m \leftarrow \emptyset$, $L^q_m \leftarrow \emptyset$,
9. For each $L_i \in Lm$ do
10. If $dist(L_i, Lp) < dist(L_i, Lq)$ then
11. $L^p_m \leftarrow L^p_m \cup \{L_i\}$;
12. Else
13. $L^q_m \leftarrow L^q_m \cup \{L_i\}$;
14. End if
15. End for
16. Derive reference movement for each split
17. $\overrightarrow{R^p_m} \leftarrow$ deriveRefMovement(L^p_m)
18. $\overrightarrow{R^q_m} \leftarrow$ deriveRefMovement(L^q_m)
19. Replace the m^{th} pattern by new ones.
20. $P' \leftarrow P - \{(\overrightarrow{Rm}, Lm)\} \cup \{(\overrightarrow{R^p_m}, L^p_m), \{(\overrightarrow{R^q_m}, L^q_m)\}$
21. Check if L(H)+L(D|H) decreases
22. If MDL(P')<MDL(P) then
23. $P \leftarrow P'$
24. End if
25. Until MDL(P')>MDL(P)
26. Return the set P of initial UT-patterns
27. Function DeriveRefMovement(L_K)
28. Consider each t-partition as a candidate
29. $R_k \leftarrow \{L | \forall L \in L_k\}$
30. Find one that minimizes the code length
31. Return s^{th} candidate $\overrightarrow{R^s_k}$, where
    $S = \underset{R^s_k \in R_k}{\operatorname{argmin}} C(\overrightarrow{R^s_k}, L_k)$
32. End function

3. Pattern Forest Construction

Input: A set P_{all} of initial UT-patterns

Output: A pattern forest FR

Processing steps:

1. $FR \leftarrow P_{all}$, $Q \leftarrow P_{all}$ where, Q is queue
2. Perform Drill-Down operation
3. While $Q \neq \emptyset$ do
4. Pop a UT-pattern UT_i from Q
5. If UT_i can be easily spilt into UT^1_i and UT^2_i then
6. Push UT^1_i and $UT2_i$ into Q;

7. Update pattern forest
8. Add two vertexes for UT^1_i and UT^2_i into FR;
9. Add two edges for ($UT_{i,}$ UT^1_i) and ($UT_{i,}$ UT^2_i) into FR;
10. End if
11. End while
12. Perform Roll-up operation
13. P_c is the set of UT-pattern in the c^{th} cluster
14. $P_c \subseteq P_{all}$ do
15. For each pair of $UT_{i,} \in P_c$ and $UT_{j,} \in P_c$ do
16. If UT_i and UT_j merged into UT_{ij} then
17. Add UT_{ij} into P_c
18. Update pattern forest
19. Add one vertex for UT_{ij} into FR;
20. Add two edges for ($UT_{ij,}$ UT_i) and ($UT_{ij,}$ UT_j) into FR.
21. End if
22. End for
23. End for
24. Return the pattern forest FR.

6. MATHEMATICAL MODEL

S :{ I, F, O}

Where, S is system

I :{ TR1, TR2…….TR_{numtra} } **where**,

TR is trajectories

F :{ F1, F2, F3, F4, F5, F6, F7, F8where,

F is set of functions

F1: Select Dataset

F2: Perform sub-trajectory clustering over I based on the partition-and-group framework

F3: Get a set C_{all} of sub-trajectory clusters

F4: Execute Initial Pattern Generation over C

F5: Get a set of P (i.e. detail levels of trajectory pattern in C_{all}) UT-patterns

F6: Construct Pattern Forest

F7: Dense trajectory identification

F8: Classification of UT-patterns such as, time restricted patterns, time delayed patterns, and time unrestricted patterns

O: {UT_1 …UT_{numpat}} **where,**

UT is Classified UT-patterns

7.EXPERMENTAL SETUP

A] Experimental Setup:

Following are the details of technologies used

Platform:

Using JAVA platform system is designed. JDK1.7 is used for JAVA environment. MySql is used for database purpose to store user basic information.

Operating Environment:

Windows OS, 4GB RAM with i3 or i3 above processor is required to run the system.

IDE:

Latest Version of NetBeans 8.0.1 is used.

B] Dataset used:

- **Starkey_OR_Main_Telemetry_1993-1996_Data [13]:**

Dataset is having trajectory details of various kinds of animals. Dataset is having Deer and Elk trajectories in 1995 (April-August) and in 1993 (May-August) respectively. Dataset is having parameters like UTMGrid, UTMGridEast, UTMGridNorth, Id, StarkeyTime, GMDate, GMTime, LocDate, LocTime, RadNum, Species, UTME, UTMN, Year, Grensunr, Grensuns, Obswt Where as main useful parameters for trajectory mining are UTMGrid, UTMGridEast, UTMGridNorth, Id, StarkeyTime, GMDate, GMTime, Species.

Where

UTMGrid = It is Universal Transverse Mercator (UTM) grid 2-dimensional Cartesian coordinate system having east and north co-ordinate details (UTMGridEast, UTMGridNorth)

Id = Animal id

StarkeyTime = Time exactly at that location

GMDate = It is date

GMTime = It is time

Species = It is code assigned to the species e.g (E for Elk and D for deer)

From this huge dataset Elk and Deer information is parsed.

A] Elk Dataset:

Dataset Description: It is having thirty three trajectories and 47,204 points. This dataset is Elk's trajectories.

B] Deer Dataset:

This dataset is having thirty two trajectories and 20,065 points.

8. RESULT TABLE AND DISCUSSION

Following figure 2, figure 3 and figure 4 represents the trajectories for individual animal dataset. Red line represents the reference trajectory lines for each cluster. Blue Line represents the dense trajectory region line where as the grey color box represents the dense trajectory movement area.

A] CATTLE DATASET:

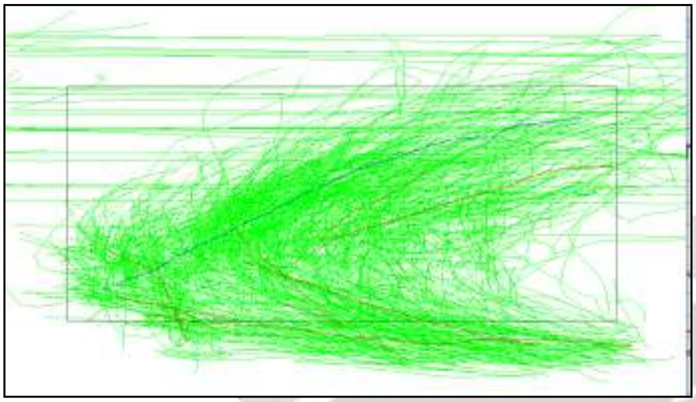

Figure 2: Sample UT-patterns for cattle dataset

B] ELK DATASET:

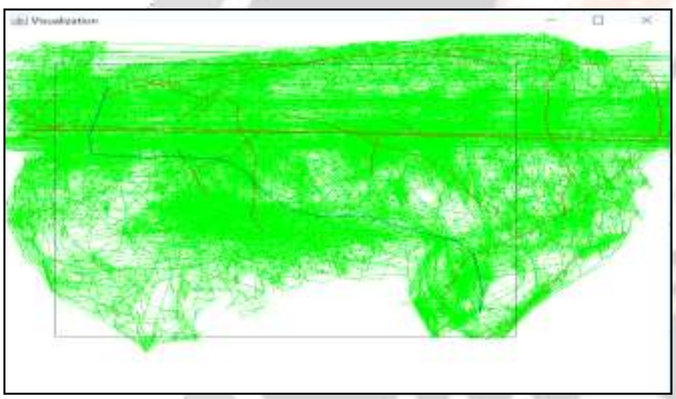

Figure 3: Sample UT-patterns for Elk dataset

C] DEER DATASET:

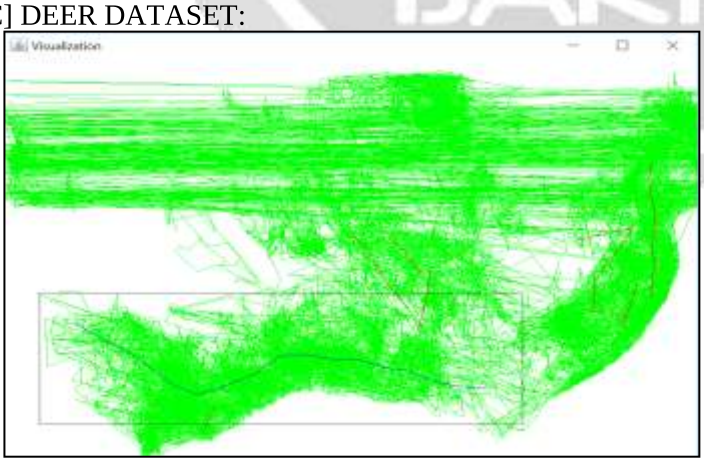

Figure 4: Sample UT-patterns for deer dataset

Patterns are classified in 2 types. Time constraint based and time independent. Time is calculated for the whole procedure. Following graph represents the time required for processing each dataset.

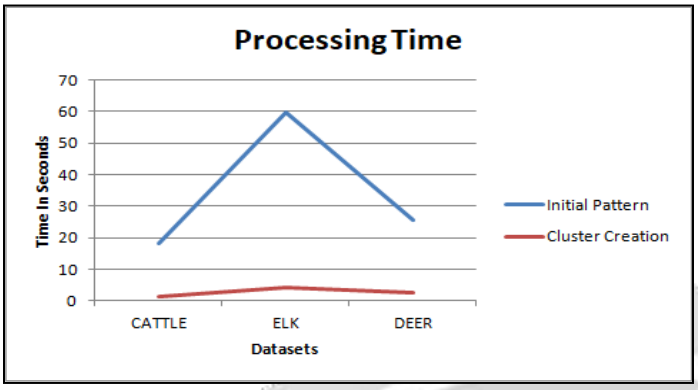

Figure 5: Graph of system processing

9. CONCLUSION

A unifying framework for mining UT-patterns has been proposed in this research work. Based on this framework, a pattern mining algorithm UT-Pattern Mine has been developed. The main advantage of the algorithm is the detection of the trajectory patterns of various temporal tightness, time-constrained, time relaxed, and time-independent. The algorithm first discovers initial UT-patterns using the intuitive information-theoretic principle of maximizing data compression and then constructs a pattern forest by drill-down and roll-up to discover more patterns. Along with trajectory pattern mining, dense trajectories are also discovered in proposed work. Experiments using real-world data sets show that UT-Pattern Mine easily discovers various types of trajectory patterns.

REFERENCES:

[1] J. Lee, J. Han and Xiaolei Li ,"A Unifying Framework of Mining Trajectory Patterns of Various Temporal Tightness", IEEE transaction on knowledge and data mining vol,27, No.6, june 2015.

[2] P. Laube and S. Imfeld, "Analyzing relative motion within groups of trackable moving point objects," in Proc. 2nd Int. Conf. Geograph. Inf. Sci., Boulder, CO, USA, Sep. 2002, pp. 132–144.

[3]P. Laube, M. J. van Kreveld, and S. Imfeld, "Finding REMO— Detecting relative motion patterns in geospatial lifelines," in Proc. 11th Int. Symp. Spatial Data Handling, Leicester, U.K., Aug. 2004, pp. 201–214.

[4] P. Laube, S. Imfeld, and R. Weibel, "Discovering relative motion patterns in groups of moving point objects," Int. J. Geograph. Inf. Sci., vol. 19, no. 6, pp. 639–668, Jul. 2005.

[5] M. Benkert, J. Gudmundsson, F. Hubner, and T. Wolle, "Reporting flock patterns," in Proc. 14th Eur. Symp. Algorithms, Zurich, Switzerland, Mar. 2006, pp. 660–671.

[6] P. Bakalov, M. Hadjieleftheriou, and V. J. Tsotras, "Time relaxed spatiotemporal trajectory joins," in Proc. 13th ACM Int. Symp. Geograph. Inf. Syst., Bremen, Germany, Nov. 2005, pp. 182–191.

[7] Sacharidis, K. Patroumpas, M. Terrovitis, V. Kantere, M. Potamias, K. Mouratidis, and T. K. Sellis, "On-line discovery of hot motion paths," in Proc. 11th Int. Conf. Extending Database Technol., Nantes, France, Mar. 2008, pp. 392–403.

[8] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: A partition-and-group framework," in Proc. ACM SIGMOD Int. Conf. Manag. Data, Beijing, China, Jun. 2007, pp. 593–604.

[9] J. Gudmundsson and M. J. van Kreveld, “Computing longest duration flocks in trajectory data,” in Proc. 14th ACM Int. Symp. Geograph. Inf.. Syst., Arlington, VA, USA, Nov. 2006, pp. 35–42.

[10] T. C. M. Lee, “An introduction to coding theory and the two-part minimum description length principle,” Int. Stat. Rev., vol. 69, no. 2, pp. 169–183, Aug. 2001

[11] Y. Li, J. Han, and J. Yang, “Clustering moving objects,” in Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Seattle, WA, USA, Aug. 2004, pp. 617–622.

[12] C. Bohm, C. Faloutsos, J.-Y. Pan, and C. Plant, “Robust informa- € tion-theoretic clustering,” in Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Philadelphia, PA, USA, Aug. 2006, pp. 65–75.
[13] http://www.fs.fed.us/pnw/starkey/data/tables/