Trajectory Segmentation and Sampling of Moving Objects Based On Representativeness

A. Evangeline Jebakumari [1] Mr. M. Gunasekaran [2]
[1]PG-Scholar, M.E-CSE [2]Assistant Professor
[1, 2]Department of Computer Science
[1, 2] AMS College of Engineering, Namakkal, India

Abstract— Moving Object Databases (MOD), although ubiquitous, still call for methods that will be able to understand, search, analyze, and browse their spatiotemporal content. In this paper, we propose a method for trajectory segmentation and sampling based on the representativeness of the (sub) trajectories in the MOD. In order to find the most representative sub trajectories, the following methodology is proposed. First, a novel global voting algorithm is performed, based on local density and trajectory similarity information. This method is applied for each segment of the trajectory, forming a local trajectory descriptor that represents line segment representativeness. The sequence of this descriptor over a trajectory gives the voting signal of the trajectory, where high values correspond to the most representative parts. Then, a novel segmentation algorithm is applied on this signal that automatically estimates the number of partitions and the partition borders, identifying homogenous partitions concerning their representativeness. Finally, a sampling method over the resulting segments yields the most representative sub trajectories in the MOD. Our experimental results in synthetic and real MOD verify the effectiveness of the proposed scheme, also in comparison with other sampling techniques.

Keywords: Trajectory segmentation, sub trajectory sampling, data mining, moving object databases.

I. INTRODUCTION

Nowadays there is a tremendous increase of Moving objects (MOD) [1] due to, on the one hand, Location-acquisition technologies like GPS and GSM networks and, on the other hand, computer vision-based tracking techniques. This explosion of information combines an increasing interest in the area of trajectory data mining and, more generally, knowledge discovery from movement-aware data [2]. All these technological achievements require new services, software methods, and tools for understanding, searching, retrieving, and browsing spatio-temporal trajectories content.

The rest of the paper is organized as follows Section II presents motivation, while Section III sets the related work. Section IV presents the methodology, proposed work is given in Section V. Finally, conclusions are provided in Section VI.

In this paper, we tackle a problem combining three different aspects. First of all, we study the problem of alternative representations of trajectories of moving objects (other than the usual sequences of 3D line segments), according to contextual information that can be automatically derived by the total trajectory population. More specifically, we investigate for an effective way to represent each trajectory by a continuous function that implicitly describes the “representativeness” of each constituent part of it with respect to the whole MOD.

Given such an intuitive representation, a second interesting arising problem is that of its segmentation in a way that an analyst could gain insight into “representative” (i.e., interesting, dense, frequent) portions (i.e., sub trajectories)but also into “non representative” parts, which are also of interest in various application scenarios . On top of the previous issues, and due to the complex nature of the trajectory data and the vast volumes of MOD, a third interesting problem arises; that of “trajectory sampling.” An insightful solution to the problem would be an analyst to be able to supervise the sampling procedure, not only regarding the volume of the sampled data set, but also the properties of the data set that reveal the underlying movement patterns of the MOD.

II. MOTIVATION

1) We propose an index-based global voting method that allows us to represent the representativeness of a trajectory in a MOD as a smooth continuous descriptor.
2) Then introduce an algorithm for the automatic segmentation of trajectories into “homogenous” sub trajectories according to their “representative-ness” in the MOD.
3) We define the problem of sub trajectory sampling in a MOD as an optimization problem and we propose a novel solution to tackle the problem.
4) Finally, conduct a comprehensive set of experiments over synthetic and real trajectory data sets, in order to thoroughly evaluate our approach.

III. RELATED WORK

In this section, we review existing works in the domains related with the current work. In our setting, representative sub trajectories are a new type of mobility pattern as such our discussion includes trajectory pattern mining, segmentation, and sampling in MOD.

A MOD consists of spatiotemporal trajectories of moving Objects (e.g., humans, vehicles, animals, etc.)In the general Case, trajectories are represented as 3D sequences where each recording encodes the 2D geographic location and the1D temporal information of mobile objects. During the last decade, several approaches have been proposed in the literature so as to enable well-known mining algorithms to operate on trajectories. One such approach is the use of different types of distance functions as the mean to group trajectories into clusters. Some approaches are inspired by the time series analysis domain [3], [7], while other exploited a set of distance operators based on primitive (space and time) as well as derived parameters of trajectories (speed and direction) [9].

An interesting approach also used in our approach is proposed in [4] for the efficient processing of most similar trajectory (MST) queries. A similar distance function is used in [8], where well-known density-based OPTICS [5] clustering algorithm, tailored to work with point data, to a new algorithm for trajectory data, named T-OPTICS (and its variant TF-OPTICS, which focuses on the discovery of the temporal intervals that lead to best clustering results). The previously mentioned temporal intervals are given by the user, so TF-OPTICS essentially re- executes T-OPTICS on segments of trajectories, obtained by properly clipping the original ones. In comparison with our approach, we automatically segment trajectories to portions based on global criteria (i.e., the representativeness of the trajectory in the MOD). Furthermore, TF-OPTICS mainly clusters whole trajectories and is not tailored to identify patterns of sub trajectories in an unsupervised way.

IV. METHODOLOGIES

A. *Global Voting Method:*

This section describes the Global Voting Algorithm (GVA). The input of the algorithm is a MOD D ¼ fT1; T2; . . . ; TNg indexed by a R-tree-like structure such as the TB-tree or the 3D-R-tree , as described in, a trajectory Tk 2 D and an intrinsic parameter _ > 0 of the method. The output of the method is the vector Vk of Lk _ 1 components that can be considered as a trajectory descriptor along the line segments ek; i 2 f1; 2; . . . ; Lk _ 1g of trajectory Tk (that Lk denotes the number of points of Tk trajectory). As such, each component of the vector Vk corresponds to the number of votes (representativeness) for each ek of Tk.

```
input : An (indexed) database
        D = {T1, T2, ··· , TN}, a trajectory Tk in
        D and the parameters σ, ε.
output: Voting vector Vk of Tk
for i = 1 to Lk − 1 do
    Vk(i) = 0
    repeat
        LoTkNN =
        Get_Next_HCNN_QT(D, ek(i), lk(i))
        V(ek(i), LoTkNN) =
        Σ∀ej∈LoTkNN,Tk≠Tj e^(−d²(ek(i),ej) / 2·σ²)
        if V(ek(i), LoTkNN) > ε then
            Vk(i) = Vk(i) + V(ek(i), LoTkNN)
        else
            break
        end
    until LoTkNN = ∅
end
```

The above algorithm provides the indexed database as input the trajectory details has been stored on the database. All the objects in the Route net will provide the details of themselves frequently. This process is a repeated one which will be refreshed once the object reached its endpoint.

B. *KNEARESTNEIGHBOR*

K-Nearest Neighbor (kNN) search is one of the most important operations in spatial DBMS, kNN queries have also been extended to moving object databases and uncertain databases.

In pattern recognition, the k-nearest neighbor algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space. K-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of its nearest neighbor.

KNN algorithm

```
Input: root, Q, α, k
Output: NN: the result set
initialize priority queue H;
enqueue ⟨root, MinDist(MQ(α), Mroot)⟩ into H;
while |NN| < k and H is not empty do
    E ← dequeue H;
    if E is an intermediate entry then
        for each child entry V of E do
            enqueue ⟨V, MinDist(MQ(α), MV)⟩;
    else if E is an object then
        add E to NN;
    else
        probe E and enqueue ⟨E, dα(E, Q)⟩;
return NN;
```

The route and object creation will take place the following attributes

1) Number of nodes
2) Number of edges
3) Speed
4) Direction
5) Signals

The process which included in the route creation and updating process

1) Update process
2) Signaling
3) Hash table

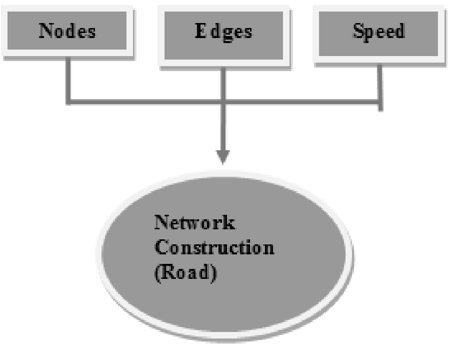

Fig. 1: Route construction process

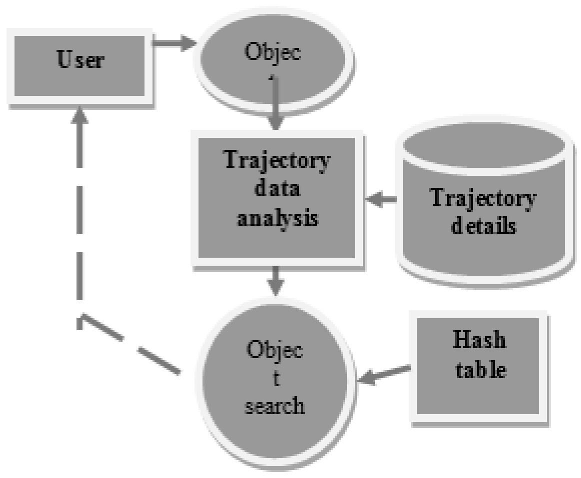

Fig. 2: Hash table technique

The above figure describes the overall process includes in the implementation. The user can search the objects from the database. The effective retrieval of data has been done using the hash table technique and KNN search method.

V. PROPOSED WORK

Traveling objects do not necessarily continuously move during a trajectory (this is the case in our example applications). Consequently, trajectories may themselves be semantically segmented by defining a temporal sequence of time sub-intervals where alternatively the object position changes and stays fixed. We call the past the moves and the latter the stops. We can then see a trajectory as a sequence of moves going from one stop to the next one (or as a sequence of stops separating the moves). For example, a bird that has departed for migration will make a stop somewhere for some time for feeding, another stop for resting, and so on till it reaches the end of its trajectory. Salespersons on a business trip will stop at all locations where they planned meeting a customer. As already stated, identifying stops (and moves) within a trajectory is the responsibility of the application. Physical stops (i.e. the fact that the position of the object is the same during two or more consecutive instants) do not account for conceptual stops, as they may be due to events that are irrelevant to the application.

The application may be interested in counting the number of stops per trajectory, and obviously the stops to be counted are only the significant stops. Hereinafter we assume that moves and stops fully cover the trajectory (i.e. there is no instant within [tbegin; tend] that belongs neither to a move nor to a stop).Traffic simulation is a dynamic problem associated with complex processes that cannot be easily described in an analytical way [1].

These processes are characterized by the interaction of several components of the system, named entities. The number of parameters is significant and the interactions are complex. Simulation models undertake to "mimic" the behavior and the interactions of real entities (cars, trucks...) in order to the structure design deals with the design of the system project. It enhances all the design of the forms of the project. The form design for each operation is studied and analyzed. The fields available for input and storage of data and information must be clear and exact for reliability. All the forms must be properly linked with the next and previous operation of the transaction. The form must have enough space and the text place to input the data and appropriate coding Thus the structured design of the proposed system can be graphically represented as:

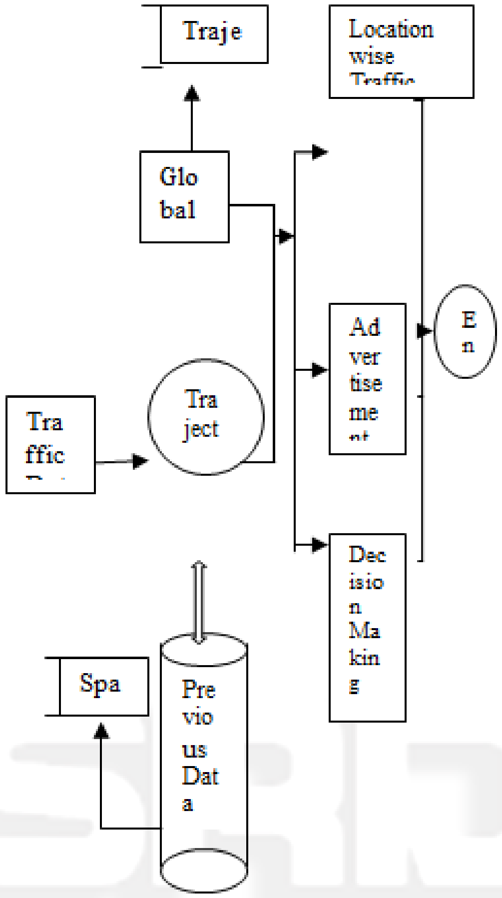

Fig 3: Proposed System

There are many methods used in the proposed work

A. *Trajectory Analysis:*

The Valuable information like traffic reports must be converted into raw trajectories for decision making purpose. When monitoring traffic, a good indication of behavior is motion. Motion is captured by trajectories which indicate the spatiotemporal characteristics of objects and encode behavior. A key observation for trajectory analysis is that typical actions are repetitive while the unusual do not occur often. This indicates that through sufficient observation one is likely to observe and can learn all the prototypical behaviors for a given scene.

B. *Location Analysis:*

Collected Raw Trajectories represent time stamped geographical locations. Apart from storing raw data in the moving object database it needs to reconstruct the trajectories. Raw points arrive in bulk sets, it needs a filter that decides if the new series of data is to be appended to an existing trajectory or not. The road users too need good quality traffic information in order to plan and adjust their routes. Traffic information has traditionally been collected with inductive-loop detectors. Location updates may help to Monitor accurate location analysis.

C. Density Analysis:

The density analysis can be used to find the gap between the vehicles and the related details. The average density gap can be found out for deciding the movement of the object. All the details can be clustered by using 'K' Means clustering. The data being collected by the participating vehicles is immediately communicated to a central facility for processing. This approach allows the collection of traffic data across the whole road network, including towns, cities, rural roads and currently unmonitored motorway segments.

Time Graph:

The time graph can be used to find the traffic details at the particular location in the particular time. This time details can be stored in the moving object database. It can be used to find the traffic of the location that the object needs to move. This graph is useful for the object to find the traffic at particular time and decide its movement.

VI. CONCLUSION

We have discussed the problem of finding representative sub trajectories in a MOD. Especially we have addressed this issue by segmentation and sub trajectory sampling based on global spatio temporal similarity of trajectories. We have proposed three algorithms GVA, SSA &TSA for trajectory voting, segmentation and sub trajectory sampling.TSA automatically estimates the number of sub-trajectories and their borders separating each trajectory of MOD into homogeneous partitions concerning their representativeness.

REFERENCES

[1] R.H. Guting and M. Schneider, Moving Object Databases. Morgan Kaufmann Publishers, 2005.
[2] F. Giannotti and D. Pedreschi, Mobility, Data Mining and Privacy, Geographic Knowledge Discovery. Springer-Verlag, 2008.
[3] M. Vlachos, D. Gunopulos, and G. Das, "Rotation Invariant Distance Measures for Trajectories," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 707-712, 2004.
[4] E. Frentzos, K. Gratsias, and Y. Theodoridis, "Index-Based Most Similar Trajectory Search," Proc. Int'l Conf. Data Eng. (ICDE), 2007.
[5] M. Ankerst, M.M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering Points to Identify the Clustering Structure," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), 1999.
[6] A. Anagnostopoulos, M. Vlachos, M. Hadjieleftheriou, E. Keogh, and P.S. Yu, "Global Distance-Based Segmentation of Trajectories," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 34-43, 2006.
[7] L. Chen, M.T. O ¨ zsu, and V. Oria, "Robust and Fast Similarity Search for Moving Object Trajectories," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), pp. 491-502, 2005.
[8] M. Nanni and D. Pedreschi, "Time-Focused Clustering of Trajectories of Moving Objects," J. Intelligent Information Systems, vol. 27, no. 3, pp. 267-289, 2006.
[9] N. Pelekis, I. Kopanakis, G. Marketos, I. Ntoutsi, G. Andrienko, and Y. Theodoridis, "Similarity Search in Trajectory Databases," Proc. Int'l Symp. Temporal Representation and Reasoning (TIME), pp. 129-140, 2007.
[10] M. Benkert, J. Gudmundsson, F. Hubner, and T. Wolle, "Reporting Flock Patterns," Proc. Conf. Ann. European Symp. (ESA), pp. 660- 671, 2006.
[11] Y. Li, J. Han, and J. Yang, "Clustering Moving Objects," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 617-622, 2004.
[12] N. Pelekis, I. Kopanakis, E.E. Kotsifakos, E. Frentzos, and Y. Theodoridis, "Clustering Trajectories of Moving Objects in an Uncertain World," Proc. Int'l Conf. Data Mining (ICDM), 2009.