International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT21719

Globular Gaussian Essential Part in Meager Bayesian Knowledge Structure for Nonlinear Deterioration

Arun J[1], Gokulakrishnan V[2]

[1]Head of the Department, Department of Computer Science and Engineering, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamil Nadu, India

[2]Assistant Professor, Department of Computer Science and Engineering, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamil Nadu, India

Article Info
Volume 7, Issue 1
Page Number: 60-67
Publication Issue :
January-February-2021

Article History
Accepted : 06 Jan 2021
Published : 16 Jan 2021

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 Databases (MOD) 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. All these technological achievements require new services, software methods, and tools for understanding, searching, retrieving, and browsing spatiotemporal trajectories content. 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 (i.e., a segment) w.r.t. 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., subtrajectories), but also into "nonrepresentative" parts, which are also of interest in various application scenarios (for example, in detecting movement outliers). 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." This is a very challenging problem where very limited work has been carried out so far. 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. In this paper, we argue that this problem can be effectively tackled if interconnected to the previous two discussed problems. In other words, we propose an automatic method for subtrajectory sampling based on the "representativeness" of the subtrajectories. In this approach, an analyst may request the top-k representative subtrajectories that best describe the MOD in an optimized way, where optimization is with respect to the "representativeness." an example of a MOD comprised by four trajectories ðfT1; . . . ; T4gÞ and the top-2 representative subtrajectories ðfS1; S2gÞ that best describe the MOD.

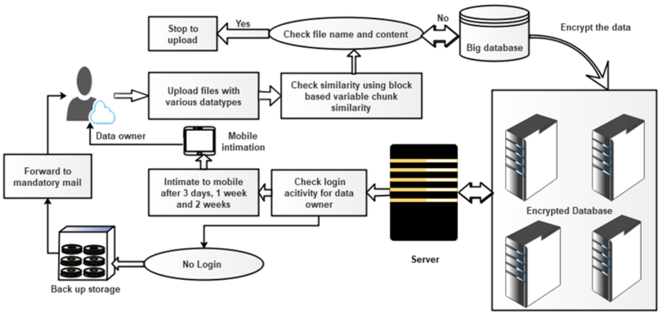
CHARACTERISTICS

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 the 1D 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 , while other exploit on a set of distance operators based on primitive (space and time) as well as derived parameters of trajectories (speed and direction) . An interesting approach also used in our approach is proposed in for the efficient processing of most similar trajectory (MST) queries. A similar distance function is used in , where Nanni and Pedreschi adapt the well-known density-based OPTICS 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 reexecutes 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 subtrajectories in an unsupervised way. Recently, Pelekis et al. proposed another approach, called CenTR-I-FCM, taking into advantage of local patterns in time dimension as the base to identify global clusters of whole approximate/symbolic trajectories. In comparison with the current work, this approach also utilizes a global but static and predefined temporal segmentation of trajectories. In addition, in this work trajectories are symbolically represented as intuitionistic fuzzy vectors and not as sequences of 3D line segments. This approach also aims at clustering trajectories as a whole with special care for handling uncertainty.

SETTING THE SCENE

In this section, we set the scene of the various aspects of the problem that this paper addresses, and concurrently we present the stepping stones where our subsequent developments base on. Let us assume an MOD D ¼ fT1; T2; . . . ; TNg of N trajectories, where Tk denotes the kth trajectory of the data set, k 2 f1; 2; . . .; Ng. We assume that the objects are moving in the xy plane. Let be the ith point, i 2 f1; 2; . . . ; Lkg, of kth trajectory, where Lk denotes the number of points of kth trajectory. denote the 2D location and the time coordinate of point pkðiÞ, respectively. Similar to the work of [25], we consider linear interpolation between successive sampled points pk þ 1Þ, so that each trajectory consists of a sequence of 3D line segments ek ¼ pkÞpki þ 1Þ, where each line segment represents the continuous moving of the object during sampled points. The goal of this work include . the automatic segmentation of the given trajectories Tk, into "homogenous" subtrajectories according to their "representativeness" in MOD and . sampling of the most representative subtrajectories of the MOD. a scheme of the proposed system architecture. The following three sections formalize the issues of trajectory representativeness, trajectory segmentation, and subtrajectory sampling, respectively. summarizes the symbols' definitions used in this work. In this research, the "representativeness" in MOD is defined by extending the definition of density biased sampling in point sets for trajectory segments. According to DBS, the local density for each point of the set is approximated by the number of points in a region, divided by the volume of the region. In our case, the "representativeness" of a trajectory segment is defined by the number of the objects that follow this segment along with time, space, and direction. Technically, "representativeness" is calculated by a voting process that is applied for each segment ekðiÞ of the given trajectory Tk, improving a preliminary version of the proposed method, presented in [24]. Thus, ek will be voted by the trajectories of MOD, according to the distance of ekÞ to each trajectory. The sum of these votes is related to the number of trajectories that are close and similar to ek, having the number of trajectories in MOD as upper limit. We avoid to give each segment the ability of voting, because, in such a case, long trajectories moving around the same area could vote many times. Moreover, under this definition, voting has the physical meaning of how many objects comove (i.e., colocation and coexistence) for a period of time. Thus, the voting results will be used to detect the representative paths and subtrajectories.


Stop to upload
Yes
Check file name and content
No
Big database
Encrypt the data
Upload files with various datatypes
Check similarity using block based variable chunk similarity
Data owner
Mobile intimation
Forward to mandatory mail
Intimate to mobile after 3 days, 1 week and 2 weeks
Check login acitivity for data owner
Server
Encrypted Database
No Login
Back up storage

II. METHODS AND MATERIAL

In this section, the proposed methodology is presented, consisting of the trajectory voting, the rajectory segmentation,and the subtrajectory sampling methods

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 ekiÞ; i 2 f1; 2; . . . ; Lk _ 1g of trajectory Tk (recall from Section 3, that Lk denotes the number of points of Tk trajectory). As such, each component of the vector VkiÞ corresponds to the number of votes (representativeness) for each ekiÞ of Tk. According to the problem formulation presented in Section 3, for each line segment ekiÞ of Tk, the proposed GVA algorithm incrementally identifies the NN segments of other trajectories Tj 2 D; j 6¼ k. For each set of NN segments represented by the list of segments/triplets in LoTkNN, the distances dðekðiÞ; ejÞ from the corresponding segments are computed. These distances are used to define the voting function V ðekðiÞ; LoTkNNÞ, which quantifies the representativeness of the line segment to a LoTkNN. In the literature, a lot of voting functions have been proposed, like step functions or continuous functions . In this work, we have selected to use the continuous function of a Gaussian kernel, which is widely used in a variety of applications of pattern recognition [29]. Formally, Note that for data collected from GPS devices where the segments are very small (due to the high sampling rate), in practice the previous function degenerates to the computation of a single Gaussian kernel, as the HCNN of a segment results in a LoTkNN containing only one NN. This is actually verified in our experiments where we used real GPS data sets. However, in cases where the data sets are highly compressed (in applications where storage cost is important), e.g., by an approach like the one proposed in [30], this may have an influence in the smoothness of the VkðiÞ descriptor. We leave such a study as future work. The control parameter _ > 0 shows how fast the function ("voting influence") decreases with distance. Given the previous assumption, and according to (4), it holds that 0 _ V ðekðiÞ; LoTkNNÞ _ 1. If dðekðiÞ; ejÞ is close to zero, the voting function gets its maximum value, i.e., 1. This means, that there exists a line segment of Tj that is being (in time, space, and direction) very close to ekðiÞ. Otherwise, if dekiÞ; ej is high, e.g., greater than 5 _ _, the voting function results in almost 0, meaning that Tj is very far away from ekðiÞ. The use of a continuous voting function, like the Gaussian kernel, gives smooth results for small changes on parameters (_ in our case), and the possibility to get decimal values as results of voting process increasing the robustness of the method. However, _ depends on space units the object movements of MOD and it is difficult to tune it. We have solved this problem by estimating _ as the percentage (e.g., 0.1 percent) of data set diameter (maximum space distance). This percentage can be kept almost constant for every data set. Finally, VkðiÞ is computed by getting the sum of votes for all of the nearest neighbor segments of trajectories Tj 2 D; j 6¼ k, according to GVA

Trajectory Segmentation Method

Having presented the voting procedure in the previous section, the next step is to provide a solution to the trajectory segmentation problem defined in Section 1.2. For this purpose, we propose the Trajectory Segmentation Algorithm (TSA) The input of the algorithm is the normalized trajectory voting signal Vk, and two intrinsic parameters w, _ of the method. The normalization is done by dividing Vk by the maximum over all Vk, thus bounding Vk _ 1. The output of the method is the segmentation Pk of Tk into LPk partitions, where LPk is automatically estimated by the proposed scheme. The method uses two sequential sliding signal windows W1 and W2 of w samples estimating the sample when the "difference" between the two windows is maximized. This methodology has been successfully applied on sound signal segmentation and P Phase Picking of seismic signals .To facilitate the discussion, two sequential sliding windowsW1 (light gray horizontal lines) and W2 (heavy gray vertical lines) locating at sample n on the given voting signal Vk. First, as the two windows slide, the two means m1, m2 and two variances _21 ,_22 of two sequential signal windows and W1, W2 locating at sample n are estimated, respectively, The next equations define m1 and _21 ; m2 and _22 are similarly defined in window W2

The first statement ensures that the difference of two windows W1, W2 will be high enough while the second selects the sample n, where dðnÞ is locally maximized, meaning that the difference of two windowsW1,W2 is locally the highest. Both statements are related with the number of partitions LPk, while the second ensures that the minimum partition size is w samples (w 3D line segments) [32]. Parameter w sets the minimum size of a partition, so w depends on the given data set and the user preferences. In other words, w expresses the minimum number of line segments that can define a subtrajectory. In addition, w is analogous to sampling rate of the trajectories (e.g., if a MOD has double sampling rate, then w should be multiplied by 2). TSA needs to estimate mean and variance measures; thus, we need at least two samples ðw _ 2Þ. Low values on w can affect the robustness of mean and variance estimation yielding false alarms (oversegmentation). High values on w gives more robust results, and it will affect the results of the method, only if there is a subtrajectory with length less than w that will not be detected. According to our experiments, when w 2 ½5; 15_ most of subtrajectories were robustly detected without important false alarms. Regarding _, it should be a positive number close to zero, in order to be sure that TSA will detect all the subtrajectories. According to our experiments, when _ 2 ½0:001; 0:1_ most of subtrajectories were detected without important false alarms, since this parameter is related with the segmentation sensitivity of our method. As _ increases, the number of subtrajectories reduces. It holds that _ can be set as a positive number close to zero (e.g., 0.01), due to the fact that in the first step, we perform normalization by dividing Vk by the maximum over all Vk, thus bounding Vk _ 1.

Subtrajectory Sampling

In the previous sections, we have presented our methodology for segmenting the trajectories of a MOD into subtrajectories using the votes gathered for the MOD. In this section, we exploit on this knowledge in order to select the top representative subtrajectories to be the result of a sampling process. In particular, we propose the Subtrajectory Sampling Algorithm (SSA). The input of the algorithm is the set of subtrajectories of the MOD Pk as estimated by TSA, the voting VPkðiÞ and the normalized lifespan NlkðiÞ vectors of the trajectory segments. The output of the method is the subtrajectory sampling set S consisting of M samples. M can be given as input to the method or (more interestingly) it can be automatically estimated by the proposed scheme The goal of SSA is the maximization of the number of subtrajectories SRðSÞ of the original MOD that are represented in the sampling set (see (3)). The complexity of an exhaustive algorithm that would search for all the possible solutions in order to maximize (3) is O N M _ _ . On the other hand, our proposed algorithm suboptimally solves the problem in ON _MÞ iterations by applying iterative optimization. SSA starts with an empty sampling set (SkðiÞ ¼ 0), where SkðiÞ is defined in Section 3.3. In each iteration step, SSA adds in sampling set an unselected subtrajectory of MOD that maximizes (3). This is equivalent with the maximization of SRSÞ gain SRgaink; iÞ (see (2)). Recall that SRgainðk; iÞ expresses the gain of SRðSÞ if we add in sampling set the ith subtrajectory of kth trajectory of the MOD. According to the proposed algorithm, it holds that

SRðSÞ gain is a monotonically decreasing function as sampling size increases. Since c VPkðiÞðjÞ _ 0,and recalling (2), it holds that

Computational Complexity Issues

Concerning the complexity of GVA, and given the use of the R-tree-like structures, the computational cost for each line segment ekðiÞ, of Tk is Oðlogð _ L _ NÞÞ, where _ L denotes the mean number of trajectory points. Executing GVA for each trajectory of the database, the total computation cost is Oð _ L _ N _ logð _ L _ NÞÞ. Concerning the complexity of TSA, the computational cost for the segmentation of a trajectory Tk is OðLkÞ, since the mean and the variance of a sliding window can be estimated recursively in Oð1Þ by the mean and the variance of the sliding window of the previous step. For example, let m1 be the mean of the window W1 ¼ Vkðn _W : n _ 1Þ and _ m1 be the mean of the next window _W1 ¼ Vkðn _W þ 1 : nÞ. Then, it holds that Conclusively, the most computationally intensive part of the proposed method is the GVA with Oð _ L N _ logð _ L _ NÞÞ complexity. In turn, the most time consuming step in GVA is the search of the nearest neighbors of a trajectory in a given time period. In order to make the application of our approach feasible to large data sets, we have adopted efficient Continuous Nearest Neighbor query processing techniques [26], where trajectories are indexed by R-treelike structures. Scalability experiments under various cases for such queries have been presented in, where it has been shown their applicability in large data sets, with an almost linear behavior with the size of data set. Actually, this conclusion is in accordance with the above theoretical analysis of the computational complexity of the proposed method.

On the Effect of the MOD Extension

As already mentioned, the proposed method is deterministic, which implies that different invocations for a given MOD will have the same result. This is a crucial and distinct characteristic of our approach w.r.t. other sampling approaches. In this section, we discuss the behavior of the proposed methods when the MOD is extended either in space or time dimensions. More specifically, we study the following scenario: In a given data set S, we add trajectories which come from a different spatial or temporal space (extension of the MOD). The question is whether the GVA,TSA, and SSA are affected by such an extension? According to this scenario, the results of the voting procedure (GVA) will not change concerning the given data set S. The new trajectories do not affect the trajectory descriptors of the trajectories of S, since they exist in different spatial or temporal space (see (4)). Therefore, the results of TSA will be exactly the same concerning the given data set S. Similarly, the new sampling set will contain the same subtrajectories as the sampling set of S (when the algorithm terminates if SRgain is lower than a given threshold), with some additional samples selected from the new trajectories, since the input of the SSA algorithm concerning S remains the same. In other words, the sampling set of the union of two distinct (in space and/or in time dimension) MOD, is the same with the union of the sampling sets, when the sampling process is performed to each MOD independently. Therefore, the effectiveness of the proposed method is not affected by the extension of the MOD.

III. CONCLUSION

In this paper, we have discussed the problem of finding representative subtrajectories in a MOD. Especially, we addressed this issue by segmentation and subtrajectory sampling based on global spatiotemporal similarity of trajectories. In particular, we have proposed three algorithms: GVA, TSA, and SSA for trajectory voting, segmentation, and subtrajectory sampling, respectively. GVA extends the density biased sampling from point sets to trajectory segments providing a local trajectory descriptor per line segment that is related to line segment representativeness.

Next, TSA automatically and effectively estimates the number of subtrajectories and their borders, separating each trajectory of MOD into homogenous partitions concerning their representativeness. Finally, SSA is applied over the resulting partitions providing the most representative subtrajectories of the MOD, also taking into account that high density regions of theMODshould not be oversampled. SSA can be automatically terminated by thresholding the number of moving objects of the original MOD that are represented in sampling set SRðSÞ. Moreover, the indexbased voting algorithm, which is the computationally most expensive step in our framework, and the polynomial computational cost of the proposed algorithms makes the scheme applicable to large databases. In our approach, contrary to related work, the temporal dimension of the MOD is taken into consideration, while there is not any inherent constraint on subtrajectory complexity and shape, yielding trajectory segmentation and subtrajectory sampling that are related only to representativeness. We have evaluated the proposed method under real and synthetic databases, and the experimental results show the effectiveness and robustness of the proposed scheme. As future work, we plan to investigate the applicability of the proposed method for (sub)trajectory clustering. The idea is that MOD clustering can be provided concurrently with MOD sampling. It holds that each subtrajectory of the sampling set has been voted by different subtrajectories of the MOD (cluster), under the minimization of the objective function proposed in the current work. Therefore, each subtrajectory of the sampling set can be considered as a cluster representative (i.e., a seed around which a cluster is formatted). This is a different tactic as the one followed in . In the same context, outliers can be discriminated from low values in voting subtrajectory descriptor (Vk).

IV. 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. Hadjieleftheriou, G. Kollios, V. Tsotras, and D. Gunopulos, "Efficient Indexing of Spatiotemporal Objects," Proc. Int'l Conf. Extending Database Technology (EDBT), 2002.

[4]. J. Han, J. G. Lee, and K. Y. Whang, "Trajectory Clustering: A Partition-and-Group Framework," Proc. ACM SIGMOD Int'l Conf.Management of Data (SIGMOD), pp. 593-604, 2007.

[5]. J.G. Lee, J. Han, X. Li, and H. Gonzalez, "Traclass: Trajectory Classification Using Hierarchical Region-Based and Trajectory-Based Clustering," Proc. VLDB Endowment, vol. 1, pp. 1081-1094, 2008.

[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.

Cite this article as :

Arun J, Gokulakrishnan V, "Globular Gaussian Essential Part in Meager Bayesian Knowledge Structure for Nonlinear Deterioration", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 1, pp. 60-67, January-February 2021. Available at doi : https://doi.org/10.32628/CSEIT21719
Journal URL : http://ijsrcseit.com/CSEIT21719