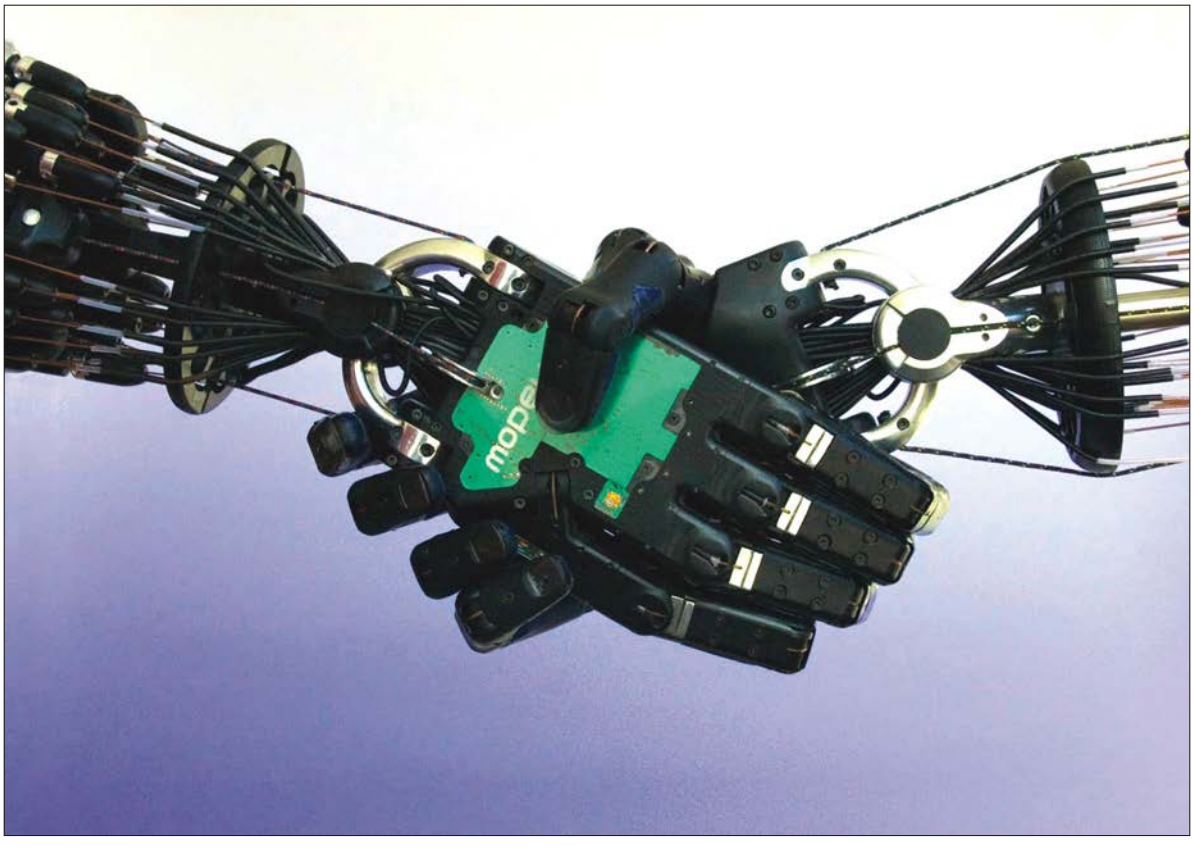

Journal of Artificial Intelligence

ISSN 1994-5450

science alert

ANSInet
an open access publisher
http://ansinet.com

OPEN ACCESS Journal of Artificial Intelligence

ISSN 1994-5450
DOI: 10.3923/jai.2017.42.48

Research Article
ONF-TRS: On-line Noise Filtering Algorithm for Trajectory Segmentation Based on MDL Threshold

Musaab Riyadh, Norwati Mustapha, Nasir Sulaiman and Nurfadlina Binti Mohd Sharef

Faculty of Computer Science and Information Technology, UPM University, Malaysia

Abstract

Background: Spatial trajectories suffer from noise that may be caused by poor signal of GPS devices, sometime the noise is acceptable few meters from its true location. In different situations, the noise is too big that dramatically change the information derive from trajectory segments such as speed, thus filtering of noise is needed before starting mining task. **Materials and Methods:** The proposed algorithm on-line noise filtering for trajectories segmentation ONF-TRS segments trajectory points to set of significant points after removing non-significant and noise points. The key idea is both non-significant and noise points have small value of (region/length), which mean travel long distance and cover small region. The threshold value of (region/length) is estimated using minimum description length concept. **Results:** Experimental results in real data sets confirm the effectiveness of (ONF-TRS) algorithm in filtering noise points during segmentation process, while existing algorithms need to implement noise filtering step before segmentation. **Conclusion:** This study provides ONF-TRS algorithm appropriate for trajectories segmentation and spatial noise filtering simultaneously which makes the algorithm convenient for stream data mining.

Key words: Spatial trajectories, trajectory segmentation, noise filtering, minimum description, length, spatial distance, moving object, characteristic points

Received: August 12, 2016 Accepted: November 15, 2016 Published: December 15, 2016

Citation: Musaab Riyadh, Norwati Mustapha, Nasir Sulaiman and Nurfadlina Binti Mohd Sharef, 2017. ONF-TRS: On-line noise filtering algorithm for trajectory segmentation based on MDL threshold. J. Artif. Intel., 10: 42-48.

Corresponding Author: Musaab Riyadh, Faculty of Computer Science and Information Technology, UPM University, Malaysia

Competing Interest: The authors have declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.

INTRODUCTION

In recent year, Moving Object Databases (MOD) have rapidly increased due to the tremendous prevalence of geolocation devices such as GPS, mobile, motion sensors and RIFD[1]. Analysis and extracting the knowledge from moving object trajectories help the researchers to understand behaviors of moving objects in the past and future. For example, the future development of a hurricane would be one of the most critical information for aid agencies. One of the important task to analysis trajectories database is segmentation, segmentation is dividing a trajectory to a set of homogenous segments according to some criteria to mine richer knowledge from local areas and minimize the complexity of computational[2]. Ideal segmentation process must maintain two desirable properties: Accuracy and concision. Accuracy means that the difference between a trajectory and its trajectory partitions should be as small as possible, while concision can be defined that the number of trajectory partitions should be as small as possible[3]. Due to the contradiction between the accuracy and concision, a typical tradeoff between the two properties must be found. Several previous researchers have addressed on trajectories segmentation, Douglas and Peucker[4] suggested an algorithm to identify the key points in maintaining a trajectory's shape. The algorithm starts with replacing the original trajectory with approximate line between the first and end points of trajectory. If the replacement meet the given error requirement, it consider the approximate line good representative for the trajectory. Otherwise, it recursively partitions the trajectory into two sub-trajectory by selecting the point that has the most errors as the split point. This process continues till the error between the approximated trajectory and the original trajectory is below the given error threshold. Douglas-Peucker's algorithm is a batch algorithm that means it needs to store all trajectory points before starting the segmentation. Lee *et al.*[3] proposed a partition-and-group framework (TRACLUS) for clustering trajectories, the partition phase divide trajectories at every characteristic point. Characteristic points are determined when partition cost function exceed the non-partition function[5,6]. These cost functions (partition and non-partition) are based on the concept of Minimum Description Length (MDL) which consists of two component L(H) and L(D|H), L(H) measures the degree of conciseness, while L(D| H) measures of the preciseness. The TRACLUS is noise sensitive algorithm. Similarly, unsupervised algorithm GRASP-UTS is proposed by Soares Junior[7] to segment trajectories from land-marks. This algorithm has the ability to modify the land-marks locations (i.e., delete, insert and change) to reach maximum homogeneity in the segments. The homogeneity measure of segments is based on the concept of Minimum Description Length (MDL). The GRASP-UTS is greedy and noise sensitive algorithm. Some researchers[8,9] have proposed segmentation algorithms based on the homogeneity inside a single trajectory, while other researchers have used homogeneity between different groups of trajectories[10].

On the other hand, spatial trajectories are never perfectly accurate, due to the sensor noise and poor positioning signals. Sometimes, the error is acceptable (e.g., a few GPS points of a vehicle fall out of the road the vehicle was actually driven), which can be fixed by map-matching algorithms. In other situations, the error of a noise is too big (e.g., several hundred meters away from its true location) to derive useful information, such as speed, therefore noise points must be removed before starting mining task. Two different approaches have been used to process noisy points in moving object trajectories[11], the first one estimates the true value of noisy points, mean (or median) filters is good example of this approach. The mean filter is preprocessing step aims to reduce noise effect by replacing the location values of each point in trajectory with the mean value of its n nearby neighbors. Knowing that the median filter is better than mean filter when handling extreme noise. The second approach removes the noisy points directly from moving object trajectory via outlier detection techniques. T-Drive[12] and GeoLife[13] projects remove trajectory segments which have speed larger than threshold.

This study proposed ONF-TRS algorithm which has the ability to remove noise points instantly during trajectories segmentation task. The algorithm overcomes the limitations of noise sensitive segmentation algorithms which consider spatial noise filtering a preceding step to trajectories segmentation. The ONF-TRS algorithm is based on MDL principle and appropriate to stream data application.

MATERIALS AND METHODS

In this study (ONF-TRS) algorithm has been proposed for on-line noise filtering and trajectory segmentation in data stream environment. The key idea of ONF-TRS is to use the (region/length) ratio of each three consecutive points (p_k, p_{k+1}, p_{k+2}) as a criteria to classify the midpoint p_{k+1} to significant, non-significant and noisy point (Fig. 1). Significant points are the points where trajectory behavior changes rapidly as illustrated in Fig. 2. The threshold value of (region/length) is estimated using the MDL concept.

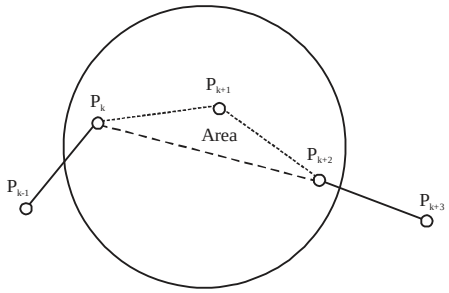

Fig. 1: Area/length ratio for three consecutive points (p_k, p_{k+1}, p_{k+2})

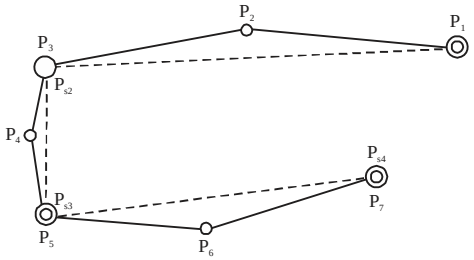

Fig. 2: Trajectory significant points P_1, P_3, P_5 and P_7

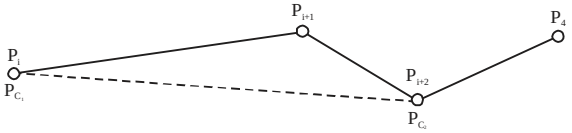

Fig. 3: Three consecutive trajectory points

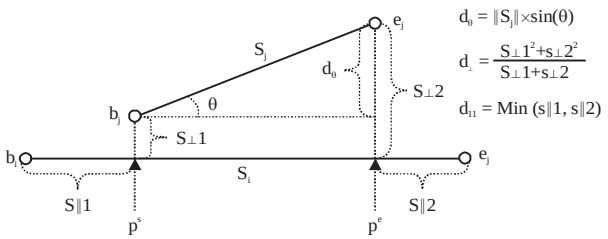

Fig. 4: Three components of the distance function

MDL concept: The Minimum Description Length (MDL) concept is widely used in information theory to describe a set of theories and applications. The key idea of this concept is for a given data-set (D) the best hypothesis (H) is the one which leads to the best compression for these data. The two basic components of MDL are: L(H) and L(D|H), "L(H) is the length, in bits, of the description of the hypothesis and L(D|H) is the length, in bits, of the description of the data when encoded with the help of the hypothesis". The best hypothesis H to explain D is the one that minimizes the sum of L(H) and L(D|H). Here, L(H) measures the degree of concision and L(D| H) that of accuracy[14]:

$$MDL = L(H)+L(D|H) \quad (1)$$

The TRACLUS[3] defined two cost functions MDL_{par} and MDL_{nopar} based on MDL concept to segment the trajectory into set of characteristic points. For three consecutive points (p_i, p_{i+1}, p_{i+2}) of a trajectory (Fig. 3), if MDL_{par} (p_i, p_{i+2})> MDL_{nopar} (p_i, p_{i+2}) then the point (p_{i+1}) which is the previous point to p_{i+2} can be classify as characteristic point otherwise (p_{i+1}) is non-significant point (can be removed). The MDL cost functions for three points (p_i, p_{i+1}, p_{i+2}) can be summarized in Eq. 2 and 3 and more detail found in the study of Lee *et al.*[3]:

$$MDL_{nopar}(p_i, p_{i+2}) = \log_2 [length(p_i, p_{i+1})+length(p_{i+1}, p_{i+2})] \quad (2)$$

$$\begin{aligned} MDL_{par}(p_i, p_{i+2}) &= \log_2 [length(p_i, p_{i+2})]+ \\ &\log_2 [(d\perp(p_ip_{i+2},p_ip_{i+1})+d\perp(p_ip_{i+2},p_{i+1}p_{i+2}))]+ \\ &\log_2 [(d\theta(p_ip_{i+2},p_ip_{i+1})+d\theta(p_ip_{i+2},p_{i+1}p_{i+2}))] \end{aligned} \quad (3)$$

The (ONF-TRS) algorithm uses the cost functions in Eq. 2 and 3 to estimate the value of (region/distance) threshold, since it processes three consecutive points in each iteration.

Distance functions: The MDL cost functions adapted three spatial distance components which are widely used in pattern recognition area[15] to measure the spatial difference between two line segments of Eq. 4-6. The three spatial component are: (1) The angle distance (d_θ), (2) The perpendicular distance ($d_\perp$) and (3) The parallel distance ($d_\parallel$) as illustrated in Fig. 4:

$$\begin{aligned} d_\theta(S_i,S_j) &= |S_j|* \sin(\theta) \text{ if } 0^\circ \leq \theta < 90^\circ \\ &|S_j| \text{ if } 90^\circ \leq \theta \leq 180^\circ \end{aligned} \quad (4)$$

where, $|S_j|$ represent the length of segment S_j and θ ($0^\circ \leq \theta \leq 180^\circ$) is the smaller angle confined between line segments S_i and S_j:

$$d\perp(S_i,S_j) = ((s\perp1)^2+(s\perp2)^2)/(s\perp1+s\perp2) \quad (5)$$

where, $s\perp1$ and $s\perp2$ are the Euclidean distance between the points (b_j, p_s) and (e_j, p_e) respectively, p_s and p_e are the projection of the points b_j and e_j onto line segment S_i.

$$d\parallel(S_i,S_j) = Min (s\parallel1, s\parallel2) \quad (6)$$

where, $s\parallel1$ and $s\parallel2$ are the Euclidean distance between the points (b_i, p_s) and (e_i, p_e) respectively. The b_i and e_i are the end points of line segments S_i.

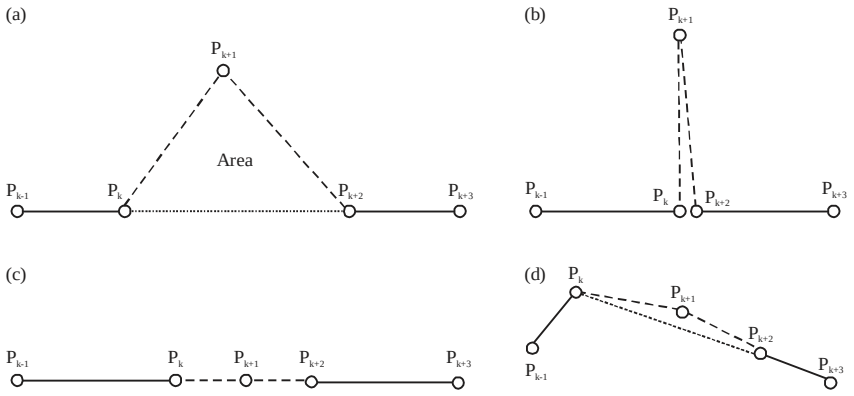

Fig. 5(a-d): Spatial pattern that could be formed from three consecutive points, (a) High value of (region/length), (b) Noisy point, (region/length) is small, (c) Collinear points in which (region/length) = 0 and (d) Non-significant point small (region/length)

Algorithm 1

Input : A set of trajectory points T = [p_1, p_2...p_j...p_{len}] each point (x, y)
Output: A set P_{sig} of significant points

01: Add p_1 to set P_{sig}
02: ptr:= 1, step:= 2; % ptr is pointer to T set
03: while (ptr+step≤len) do
04: Region= Area of polygon (p_{ptr}, p_{ptr+1}, p_{ptr+2});
05: distance= Euclidean distance between (p_{ptr}, p_{ptr+1}, p_{ptr+2});
06: ratio=Region/distance;
07: if (ratio>threshold)
08: Add P_{ptr+1} into the set P_{sig}; % p_{k+1} significant point
09: ptr= ptr+1;
10: else
11: remove P_{Ptr+1} from the set T; % p_{k+1} non-significant or noisy points
12: len = length (T)
13: end % if
14: end % while
15: Add p_{len} to set Psig

Fig. 6: ONF-TRS algorithm

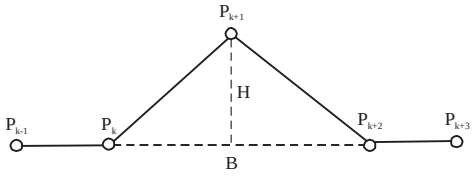

Fig. 7: Value of H such that $MDL_{par} = MDL_{nopar}$

Proposed algorithm: In (ONF-TRS) algorithm, the value of (region/length) for three sequential points of trajectory (p_k, p_{k+1}, p_{k+2}) plays key role in classifying the midpoint p_{k+1} to significant, non-significant and noisy point. Where region represents the area of the triangle (p_k, p_{k+1}, p_{k+2}) and length is the Euclidian distance between these points as illustrated in Fig. 5a. It is obvious that for any three points (p_k, p_{k+1}, p_{k+2}) the value of (region/length) is small (less than estimated threshold) in two cases: (1) Either the value of region is small because the point p_{k+1} is too close from the triangle base (p_k, p_{k+2}), here point p_{k+1} will be considered non-significant point and can be removed as illustrated in Fig. 5c and d and (2) Or the value of length is too large in case of point p_{k+1} is too far from the base of triangle (p_k, p_{k+2}), here p_{k+1} will be considered noisy point also can be removed as illustrated in Fig. 5b. Otherwise the point p_{k+1} is added to set of significant points of trajectory P_{sig} in case of (region/length) is larger than estimate threshold.

The big challenge of (ONF-TRS) is how to estimate the threshold value for (region/length) that gives high accurate segmented trajectory and noisy points filtering. Ultimately, the reason why we outweigh the ratio (region/distance) rather than MDL cost functions is its ability to get rid of non-significant and noisy points, while MDL cost functions in TRACLUS[4] get rid of non-significant point only. Figure 6 illustrates the steps of ONF-TRS algorithm.

Initially, line 1 add the first point of trajectory T into set P_{sig}, line 2 set the pointer ptr to first point of trajectory T. Lines 3-5, calculate the (region/length) value for three consecutive points starting from pointer ptr, if the value of (region/length) larger than threshold value add point p_{ptr+1} to set P_{sig} (significant point) line 8. Otherwise remove point p_{ptr+1} from set T (non-significant or noisy point) line 11. Line 15 add the end point of trajectory T into set P_{sig}.

Threshold estimation: The algorithm (ONF-TRS) uses the cost functions in Eq. 2 and 3 to estimate the threshold value of (region/length). Firstly, we compute the value of H such that $MDL_{par} = MDL_{nopar}$ that mean point p_{k+1} on edge of becoming characteristic point according to TRACLUS as illustrated in Fig. 7. Secondly, the H value is used to compute the area of

triangle p_k, p_{k+1}, p_{k+2} (region). Finally, calculate the (region/length) where length is the Euclidian distance between p_k, p_{k+1}, p_{k+2}. Figure 8 shows the steps to compute the estimated threshold.

RESULTS

In this study, the performance of ONF-TRS algorithm was assessed using two real data sets Elk 1993 and Deer 1995 data set. The Elk 93 has 33 trajectories and 47204 points, while Deer 95 has 32 trajectories and 20065 points. The tests have been conducted under the environment: Windows-8 operating system, Acer laptop computer (processor: Intel Core i7 CPU (2.20) GHz and (16 GB) RAM. Matlab R2015a and excel 2013 were used to implemented the algorithm and plot the charts.

Threshold estimation: To estimate threshold value, algorithm 2 had been implemented many times on different sets of three points that have different base length 1, 10, 100 and 1000. In each run, the value of height value (H) is calculated such that $MDL_{par}(p_k, p_{k+2}) = MDL_{nopar}(p_k, p_{k+2})$. The

Algorithm 2	
Input:	Three consecutive points in trajectory (p_k, p_{k+1}, p_{k+2})
Output:	Estimated threshold value for (region/length)
01:	Calculate H value such that $MDL_{par}(p_k, p_{k+2}) = MDL_{nopar}(p_k, p_{k+2})$
02:	Region = (length (p_k, p_{k+2})*H)/2; % area of triangle
03:	Distance= length(p_k, p_{k+1})+length(p_k, p_{k+1})
04:	Threshold=Region/distance

Fig. 8: Algorithm to estimate the approximate value of region/length

(region/length) threshold is computed based on H value as illustrated in Table 1. By using linear interpolation on threshold column of Table 1, the threshold value for ELK 93 and Deer 95 data set is approximately (0.35), since the average length between three consecutive points for these data set is nearly (400).

ONF-TRS algorithm versus TRACLUS: Two trajectories were randomly selected from data sets ELK 93 and Deer 95 to compare the result of proposed algorithm vs TRACLUS, the comparison is in term of number of segments (compression rate) as illustrated in Table 2.

The chart in Fig. 9 shows the actual number of segments for data set Deer 1995 (32 trajectories) and the number of segments after processing the data set using TRACLUS and ONF-TRS algorithms.

Table 1: Value of (region/length) for different triangle base length

Base length (B)	Height (H)	Length p_k, p_{k+1}, p_{k+2}	Region B×H/2	Threshold (region/length)
1	0.637	1.6196	0.318	0.197
10	0.677	10.0915	3.39	0.336
100	0.703	100.01	35.20	0.352
1000	0.706	1000	353	0.3530

Table 2: Percentage of segments minimization achieved by ONF-TRS algorithm compared with TRACLUS algorithm based on (0.35) threshold

Data sets	Algorithm	No. of segments	Segments minimization (%)
ELK 93	TRACLUS	1417	170 (1437-1403)/(1437-1417)
(1437 segments)	ONF-TRS	1403	
ELK 93	TRACLUS	1531	196
(1552 segments)	ONF-TRS	1511	
Deer 95	TRACLUS	1163	241
(1175 segments)	ONF-TRS	1146	
Deer 95	TRACLUS	1077	238
(1090 segments)	ONF-TRS	1059	

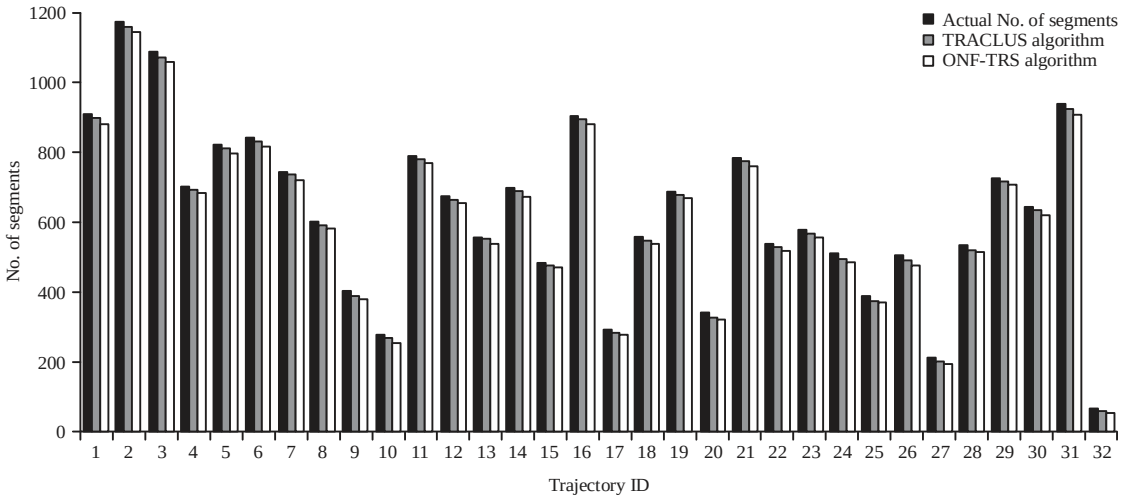

Fig. 9: Number of segments for each trajectory of data set Deer 95, No. of segments after processing the data set by TRACLUS algorithm and No. of segments after processing the data set by ONF-TRS algorithm

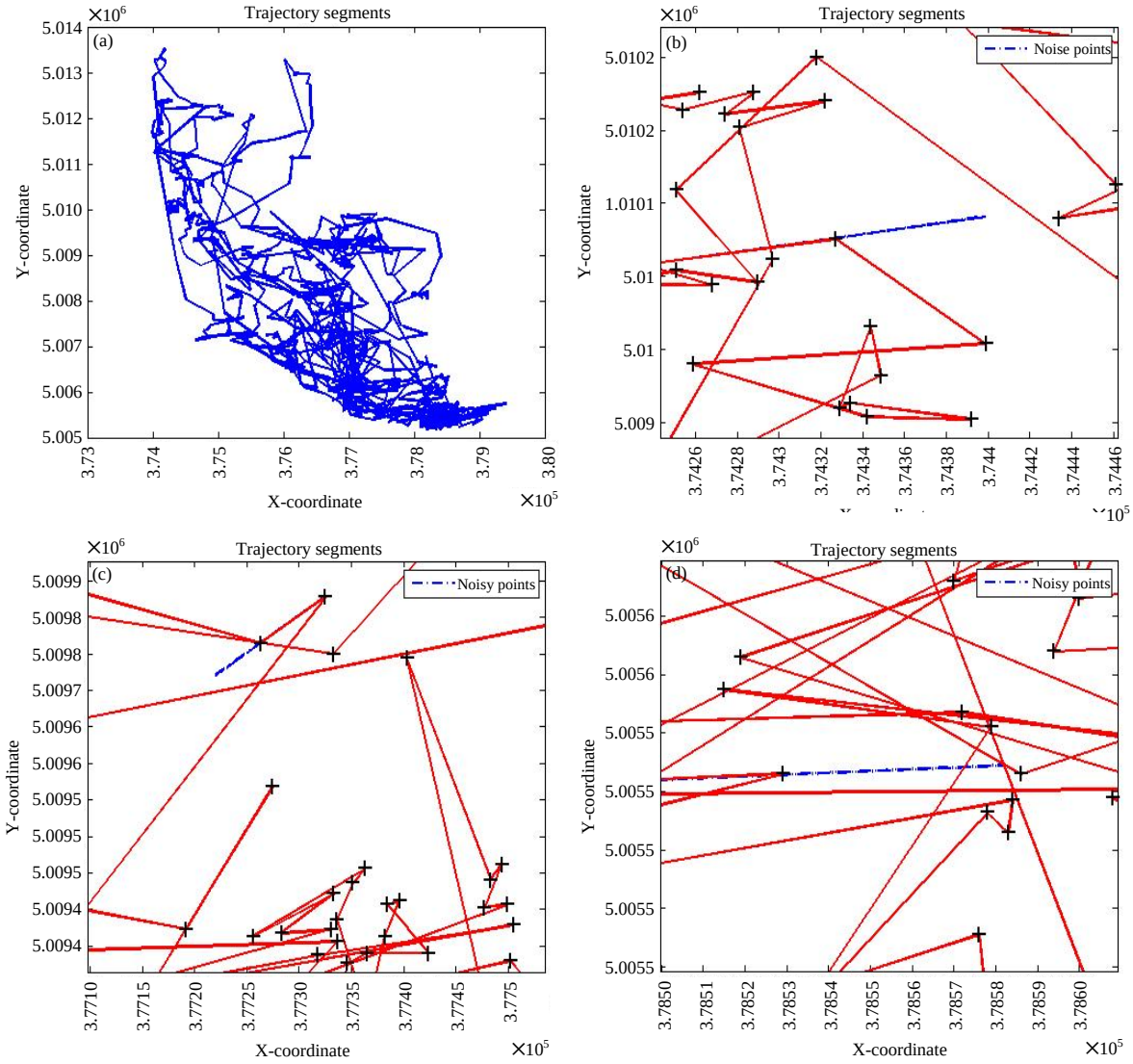

Fig. 10(a-d): Trajectory noisy points that detect and remove by ONF-TRS algorithm, (a) Whole trajectory and (b-d) Noisy points (blue lines)

The plotting of selected trajectory from Deer 95 data set is illustrated in Fig. 10a and selected noisy points are illustrated in Fig. 10b-d.

Time complexity: The time complexity of the (ONF-TRS) algorithm is O (n), where n is the number of points of a trajectory T.

Segmentation accuracy: The accuracy of segmentation process and noise filtering of (ONF-TRS) algorithm depends on threshold value, if the application need high accuracy segmentation then a small value of threshold will be chosen. The threshold (0.35) is reference value which give us segmentation process too close to TRACLUS algorithm for ELK 93 and Deer 95, but (ONF-TRS) algorithm has noise filtering feature. Table 3 illustrate the segments number for different value of threshold applied on ELK 93 trajectory with (1437 segments).

Table 3: Different values of threshold applied on ELK 93 trajectory (1437 segments)

Threshold value	No. of segments
0	1437
0.1	1418
0.35 (reference value)	1403
1	1362
2	1277

DISCUSSION

In this study, result of ONF-TRS is compared with TRACLUS, study of Li *et al.*[5] and GRASP-UTS, because all of these algorithms based on Minimum Description Length (MDL) principle. Trajectory segmentation algorithm must maintain two properties: Accuracy and concision. The proposed algorithm ONF-TRS compared with TRACLUS and study of Li *et al.*[5] improved the concision (minimize number of segments) and keep the same level of accuracy, the minimization is due to noisy points removal. The minimization percentage range between (170-241%). Furthermore, the proposed algorithm calculate typical threshold value for every dataset (e.g., 0.35 for ELK 93) dataset to maintain accuracy and concision and noisy point removal. The threshold value of proposed algorithm value can be adjusted (minimize to get higher accuracy or maximize to get higher concision) which make the ONF-TRS more flexible than TRACLUS, study of Li *et al.*[5] and GRASP-UTS to meet the requirements of (low/high) accuracy applications. Besides that, GRASP-UTS is greedy and noise sensitive algorithm which make it not appropriate for data stream application, in contrary with propose algorithm.

CONCLUSION

In this study, ONF-TRS algorithm was proposed for trajectory segmentation and on-line noise filtering. The algorithm depends mainly on the value of region/length for three consecutive points of trajectory, if the region/length value is less than estimated threshold value the second point of the three points consider non-significant or noisy point otherwise the point is significant point. The threshold value estimated based on MDL cost functions. The ONF-TRS algorithm is convenient for stream mining application.

REFERENCES

1. Lv, C., F. Chen, Y. Xu, J. Song and P. Lv, 2015. A trajectory compression algorithm based on non-uniform quantization. Proceedings of the 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, Augusst 15-17, 2015, Zhangjiajie, pp: 2469-2474.
2. Buchin, M., A. Driemel, M. van Kreveld and V. Sacristan, 2011. Segmenting trajectories: A framework and algorithms using spatiotemporal criteria. J. Spatial Inf. Sci., 3: 33-63.
3. Lee, J.G., J. Han and K.Y. Hwang, 2007. Trajectory clustering: A partition and group framework. Proceedings of the ACM SIGMOD International Conference on Management of Data, June 11-14, 2007, Beijing, China, pp: 593-604.
4. Douglas, D.H. and T.K. Peucker, 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Can. Cartographer, 10: 112-122.
5. Li, Z., J.G. Lee, X. Li and J. Han, 2010. Incremental clustering for trajectories. Proceedings of the 2010 15th International Conference on Database Systems for Advanced Applications, April 1-4, 2010, Tsukuba, Japan, pp: 32-46.
6. Zaghlool, E., S. ElKaffas and A. Saad, 2015. A density-based clustering of spatio-temporal data. Adv. Intell. Syst. Comput., 354: 41-50.
7. Soares Junior, A., B.N. Moreno, V.C. Times, S. Matwin and L.D.A.F. Cabral, 2014. GRASP-UTS: An algorithm for unsupervised trajectory segmentation. Int. J. Geog. Inf. Sci., 29: 46-68.
8. Buchin, M., H. Kruckenberg and A. Kolzsch, 2015. Segmenting Trajectories by Movement States. In: Advances in Spatial Data Handling: Geospatial Dynamics, Geosimulation and Exploratory Visualization, Springer, New York, USA., ISBN: 9783642323157, pp: 15-25.
9. Rocha, J.A.M., V.C. Times, G. Oliveira, L.O. Alvares and V. Bogorny, 2010. DB-SMoT: A direction-based spatio-temporal clustering method. Proceedongs of the 2010 5th IEEE International Conference Intelligent Systems, July 7-9, 2010, London, pp: 114-119.
10. Panagiotakis, C., N. Pelekis, I. Kopanakis, E. Ramasso and Y. Theodoridis, 2012. Segmentation and sampling of moving object trajectories based on representativeness. IEEE Trans. Knowledge Data Eng., 24: 1328-1343.
11. Zheng, Y., 2015. Trajectory data mining: An overview. ACM Trans. Intell. Syst. Technol., Vol. 6. 10.1145/2743025
12. Yuan, J., Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun and Y. Huang, 2010. T-drive: Driving directions based on taxi trajectories. Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, November 03-05, 2010, San Jose, CA, USA., pp: 99-108.
13. Zheng, Y., X. Xie and W.Y. Ma, 2009. GeoLife 2.0: A location-based social networking service. Proceedings of the 10th IEEE International Conference on Mobile Data Management: Systems, Services and Middleware, May 18-29, 2009, Taipei, China, pp: 357-358.
14. Rissanen, J., 1978. Modeling by shortest data description. Automatica, 14: 465-471.
15. Chen, J.Y., M.K. Leung and Y.S. Gao, 2003. Noisy logo recognition using line segment hausdorff distance. Pattern Recogn., 36: 943-955.