Outlier Detection in Vehicle Trajectories

Vaishali Mirge
Dr. C.V. Raman University,
Kota, Bilaspur (C.G)

Kesari Verma
Department of MCA,
National Institute of Technology,
Raipur (C.G)

Shubhrata Gupta
Department of Electrical Engineering
National Institute of Technology,
Raipur (C.G)

ABSTRACT
Outlier detection in vehicle trajectory data is an important research problem of recent era. This problem has gained attention with the development of global position system (GPS), wireless technology and location aware services, which makes possible to gather a large quantity of trajectory data. This paper presents an algorithm for anomaly detection in vehicle trajectory data using hausdorff distance. The algorithm has the capability of handling non-uniform data, data of unequal length, and data on different directions. The Proposed technique identifies anomalous trajectories and those trajectories as well which partially behave anomalous activity.

In the proposed technique the clusters of nearest trajectories are formed based on hausdorff distance. The outlier trajectories are identified based on user defined outlier threshold. If any cluster is containing less number of trajectories than the outlier threshold, the trajectories of that clusters are identified as outlier trajectories. The algorithm has been tested on real data set of School Buses [13].

Keywords
Anomalous Trajectory Patten, Outlier Trajectories, Trajectory Analysis, Trajectory Pattern Mining

1. INTRODUCTION
Recent improvements in satellites and tracking facilities have made it possible to collect a huge amount of trajectory data of moving objects such as vehicles. These tremendous amount of data helps to develop a model which detect automatically the trajectories that deviate significantly from the expected or typical trajectories. An outlier is a data object that is grossly different from or inconsistent with the remaining set of data. Trajectory outliers may be indicative of illegal and adverse activity. Timely detection of these relatively infrequent events, which is critical for enabling pro-active measures, requires careful analysis of all moving objects at all time. Thus, there is a need for automated trajectory analysis.

This paper is concerned with the problem of creating algorithms that can automatically detect suspected outlier in vehicle trajectory data. The proposed algorithm compares vehicle trajectories based on Hausdorff distance. The Hausdorff distance is used for image matching and image analysis [8], but in this paper it is used to match trajectories. In the proposed work hausdorff distance is calculated for each trajectory with every other trajectories. Further for each trajectory, cluster of nearest trajectories are formed based on hausdorff distance. If number of trajectories in a cluster of a particular trajectory is less than outlier threshold then corresponding trajectory is identified as outlier trajectory. It can be expressed using equation (1).

$$\begin{cases} 1.\ D(T_i, T_j) < \sigma \\ \textbf{Add } \mathbf{T_j} \textbf{ in the cluster } \mathbf{C_i} \\ 2.\ \text{Numer-of-Trajectories}(C_i) < \alpha \\ \mathbf{T_i} \textbf{ is Anomalous Trajectory} \end{cases} \quad \text{Eq-1}$$

In Eq-1 σ and α are similarity and outlier threshold respectively. D is hausdorff distance and C_i is clusters of nearest trajectories of T_i.

The Remainder of the paper is organized as follows : Section 2 reviews the related literature. Section 3 comprises of problem definition. Section 4 presents proposed method in detail. Section 5 explains the Out-tra and Haus-dorff-distance algorithm. In section 6 we present an experimental evaluation of proposed approach. Finally, Section 7 concludes this paper.

1.1 Dataset
Real data set, School Buses [13] consists of 145 trajectories of 2 school buses collecting (and delivering) students around Athens metropolitan area in Greece for 108 distinct days. For the experimental purpose we worked on 60 trajectories. Figure 1 shows the trajectory path of school bus data.

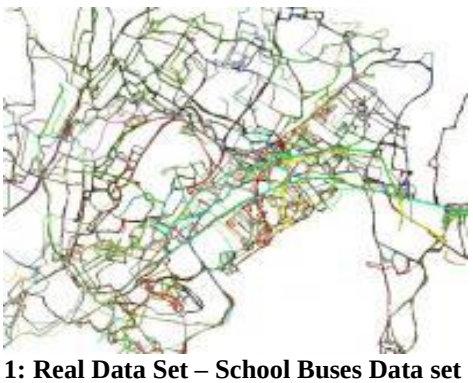

Fig 1: Real Data Set – School Buses Data set

2. RELATED WORK
An outlier is an observation that differs so much from other observations as to arouse suspicion that it was generated by a different mechanism. Despite its importance, trajectory outlier detection has not been paid the attention as it deserves. A significant work related to automated outlier detection in trajectory data has been done by Knorr et al. [1]. In their technique, a trajectory is represented as a set of key features instead of a sequence of points. A trajectory is summarized by

the coordinates of the starting and ending points; the average, minimum, and maximum values of the directional vector; and the average, minimum, and maximum velocities. The distance function is simply defined as the weighted sum of the difference of these values. Then, a distance-based algorithm is applied for detecting trajectory outliers. Giannotti et al. [2] Introduces a novel form of spatio-temporal pattern, called as a trajectory pattern, represents a set of individual trajectories that share the property of visiting the same sequence of places with similar travel times. Two notions are central to this paper: (i) the regions of interest in the given space, and (ii) the typical travel time of moving objects from region to region. Aggarwal et al. [3] has developed the technique for outlier detection which is applicable for very high dimensional data sets. The method works by finding lower dimensional projections which are locally sparse, and cannot be discovered easily by brute force techniques because of the number of combinations of possibilities. Agarwal et al. [5] explores various problems related to minimizing Hausdorff distance between sets of points, disks, and balls. It computes exactly or approximately the smallest Hausdorff distance over all possible rigid motions. Li et al. [7] proposed the ROAM framework for the problem of anomaly detection in massive moving object data sets. ROAM uses, motif-based feature space representation with automatically derived hierarchies. Combined with a rules-based classification model that explores the hierarchies. Daniel et al. [8] shows how Hausdorff distance is used to compare two binary images. The method compares a 32x32 model bitmap with a 256 x 256 image bitmap in a fraction of a second. Lee et al. [9] proposed a novel framework, the partition-and-detect framework, for trajectory outlier detection. Based on this framework, they developed a trajectory outlier detection algorithm. The main advantage of the algorithm is the detection of outlying sub-trajectories from a trajectory database. Ramaswamy et al. [10] presents a formulation for distance based outliers that is based on the distance of a point from its k^{th} nearest neighbor. It rank each point on the basis of its distance to its k^{th} nearest neighbor and declare the top n points in this ranking to be outliers. Pokrajac et al. [11] Proposed incremental LOF (Local Outlier Factor) algorithm for detecting outliers in data stream. Lee et al. [12] Developed the trajectory clustering algorithm TRACLUS. As the algorithm progresses, a trajectory is partitioned into a set of line segments at characteristic points, and then, similar line segments in a dense region are grouped into a cluster. Mirge et al. [14]–[15] introduced algorithms to mine trajectory patterns of moving vehicles in order to identify heavy traffic sections of the unidirectional and bidirectional road network.

3. PROPOSED METHOD

Let T ={T_1, T_2, T_3… T_n} be a set of trajectories. Purpose of our work is to identify outlier trajectories. Hausdorff distance is used to measure the similarity between two trajectories. Outlier trajectory is identified by generating the clusters based on hausdorff distance and applying outlier thresholds to discover outlier trajectory. Figure 2 shows the outlier trajectory which followed the unusual path.

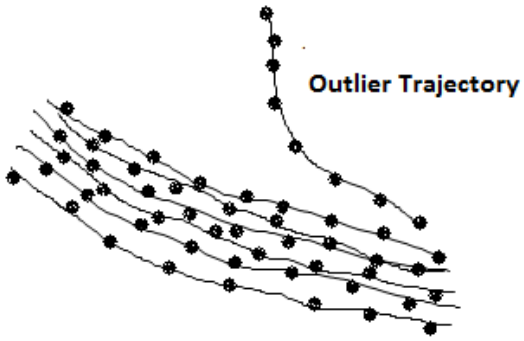

Fig 2: Outlier Trajectory

Definition1 (Hausdorff Distance)

Hausdorff distance is the *maximum distance of a set to the nearest point in the other set.* More formally, Hausdorff distance from set A to set B is a *maximin* function, defined as

$$h(A,B) = max\{_{a \in A} min\{_{b \in B} (d(a,b))$$

Where a and b are points of sets A and B respectively, and d(a, b) is any metric between these points ; for simplicity, we'll take d(a, b) as the Euclidian distance between a and b. Hausdorff distance is used for measure dissimilarity between two trajectories. Using this measure, a trajectory A is considered similar to B *iff* every point in A is close to at least one point in B. To demonstrate the effectiveness of this measure, example in Figure 3 illustrates how MINDIST(a,B) for each a in A can be evaluated by finding the nearest neighbor (NN) in B. This definition shows that HAUSDIST(A,B) is asymmetrical. The symmetrical Hausdorff distance SYMHAUSDIST(A,B) is defined as MAX{HAUSDIST(A,B), HAUSDIST(B,A)}. Figure 3 shows the hausdorff distance between two sets A and B which is equivalent to distance between the points a_4 and b_4.

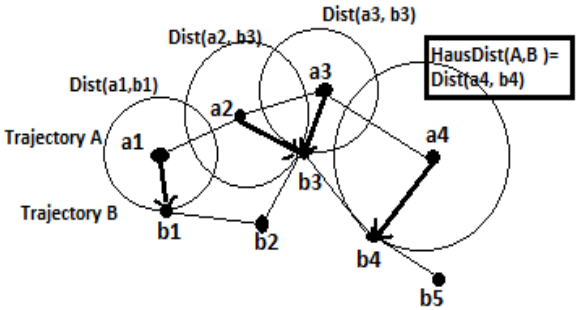

Fig 3: HAUSDORFF Distance (A, B)

In proposed work for each trajectory, cluster of all other similar trajectories are identified. Hausdorff distance is calculated for a trajectory with every other trajectories in trajectory set. Cluster of any trajectory T will hold only the trajectories whose hausdorff distance with T is lower than the similarity threshold. If number of trajectories in a cluster of trajectory T is less than the outlier threshold then T will be outlier trajectory. Same process is repeated for all the trajectories in the trajectory set. As shown in figure 4 trajectory T_1, T_2 and T_3 are close to each other. These trajectories form the cluster 1 where as trajectory T_4 is far

away from T_1, T_2, T_3. Therefore there will be cluster 2 with only one trajectory T_4 . Thus T_4 is outlier among T_1, T_2,T_3,T_4 as shown in Figure 4.

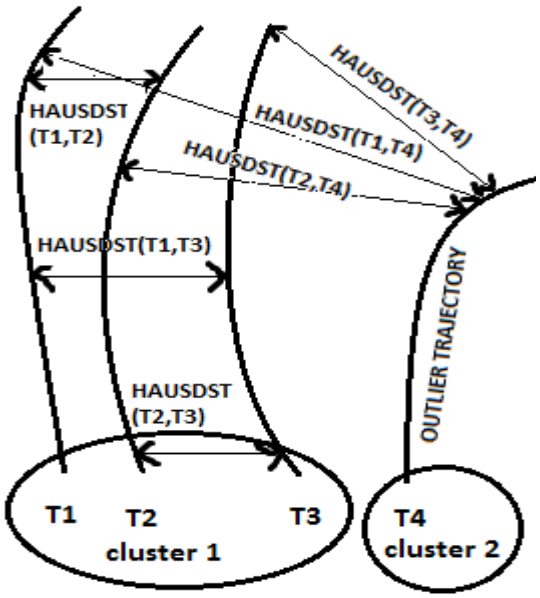

Fig 4: Detecting outlier trajectory

4. ALGORITHMIC FRAMEWORK

4.1 Outlier Trajectory Detection

Identifying the outlier trajectory requires to determine the similarity among trajectories. hausdorff distance has been used as a similarity measure between two trajectories. Proposed algorithm Out-Tra first find hausdorff distance D_{ij} for each trajectory T_i with every other trajectory T_j in the trajectory set T. if D_{ij} is less than the similarity threshold σ then corresponding trajectory T_j is added to the cluster C_i. Otherwise discarded. Same process is repeated for each trajectory in T and generates cluster C_1, C_2 … C_n corresponding to trajectory T_1, T_2,… T_n. Each cluster C_i will have the trajectories which are closest to T_i. Number of trajectories in each cluster C_i is compared with outlier threshold α. If it is lower than α, trajectory T_i corresponding to cluster C_i is recognized as outlier trajectory.

Algorithm- Out-Tra (T, σ, α): An a
lgorithm for detection of Outlier Trajectories.
Input:

T ={T_1, T_2, T_3… T_n} where Trajectory T_i = {x_{i1}, y_{i1},t_{i1}), (x_{i2}, y_{i2}, t_{i2}) … (x_{im}, y_{im}, t_{im}), id} in which id is the trajectory id. Similarity threshold σ, Outlier Threshold α.

Output :

Set of outlier trajectories : O_t

Procedure :

Begin

1. **for each** T_i where i=1 to n **do**
2. **for each** T_j where j=1 to n **do** Calculate D_{ij} ←Haus-dorff-distance (Ti, Tj)
3. **If** $D_{ij} < \sigma$
4. Add j to cluster C_i .
5. **End if**
6. **End for**
7. **End for**
8. **For** i=1 to n do
9. If no_of_elements_ in_cluster(C_i) < α
10. $O_t \leftarrow T_i$
11. **End if**.
12. **End for**

4.2 Hausdorff Distance Calculation

The Hausdorff distance is a measure of the maximum of the minimum distance between two sets of objects. Mathematically Hausdorff distance between two set A and B is defined as

$$h(A,B) = max\left\{ \begin{matrix} a \\ a \in A \end{matrix} min\left\{ \begin{matrix} a \\ b \in B \end{matrix} (d(a,b)) \right.\right.$$

Algorithm- Haus-dorff-distance: (T_i, T_j) An algorithm for calculating hausdorff distance between two Trajectories.

Input: Two trajectories T_i and T_j
Output: Hausdorff Distance D_{ij}
Begin

1. Maximum ←0
2. **For** k= 1 to m
3. minimum ← 0
4. **For** l= 1 to m
5. D_k = Euclidian-Distance((x_{ik}, y_{ik}), (x_{jl}, y_{jl}))
6. If minimum > D_k
7. minimum= D_k
8. End for
9. If minimum > maximum
10. maximum=minimum
11. End for

5. EXPERIMENT RESULT

The experiment has been performed on real data set. School [13]. It consists of trajectories of 2 school buses collecting (and delivering) students around Athens metropolitan area in Greece. For the experiment purpose part of the data set has been taken which consists of 60 trajectories.

For experimentation in our framework, we implemented the proposed algorithm in MATLAB. Experiments were performed in Pentium IV machine with 500 GB Hard disk and 1 GB RAM.

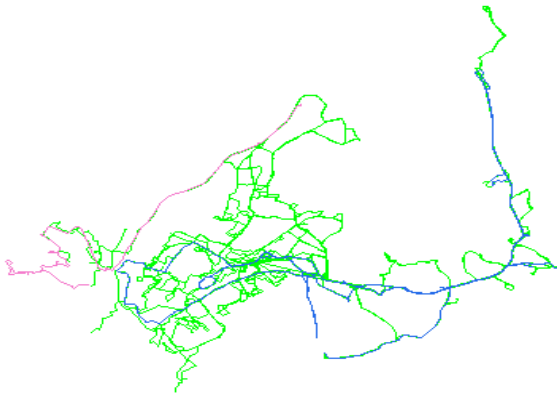

Fig 5: School Buses Data Set (Partial)

Figure 5 shows the result for a small portion of the School buses data set The parameters are set as follows: σ =10000 , α =3. Here lines in green colour representing normal trajectories, while Pink and Blue lines representing the outlier trajectories, it is automatically observed by our algorithm. When applying algorithm **Out-Tra** it separates the outlier trajectory by finding the hausdorff distance for each trajectory with every other trajectory then making cluster for each trajectory with its nearest trajectory. We can observe from the figure 5 that for blue and pink trajectories, some trajectory points are far away from the trajectory points of other trajectories. When calculating hausdorff distance between outlier trajectory with other trajectories value will be higher as compare to the hausdorff distance among other trajectories. Making cluster for pink and blue trajectories on the basis of hausdorff distance, it has only one or two trajectories which are represented as outlier trajectories.

5.1 Performance Evaluation

Table 1. Performance based on outlier (α) and similarity threshold (σ)

α (σ=10000)	3	8	13	18
No. of outliers	4	6	7	7
σ (α =3)	10000	15000	20000	25000
No. of outliers	4	4	2	0

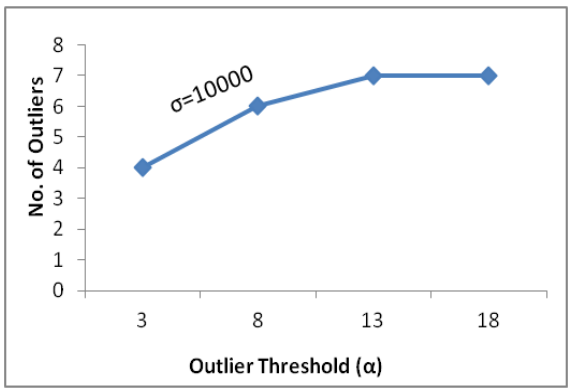

Fig 6: Outlier Threshold Vs No. of Outliers

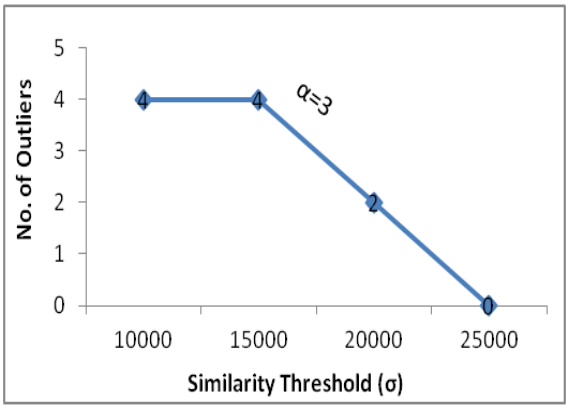

Fig 7: Similarity Threshold Vs No. of outliers

Table 1 shows the changes in No. of outliers with respect to similarity and outlier threshold. Figure 6 represents that as the outlier threshold increases, no. of outliers also increases. This is so because when increasing the outlier threshold, clusters with higher number of trajectories are also declared as outliers. When the number of outlier cluster increases it directly affects number of outliers also, due to the outlier clusters are direct relationship with outliers. Similarly as shown in Figure 7 when the similarity threshold is increased, It raised the range of distance to cover nearest trajectories. When forming clusters it will increase the number of trajectories in a cluster. We can conclude that there will be less chances of getting clusters with lower number of trajectories which reduces the number of outlier trajectories also.

5.2 Accuracy of the Result

Table 2. Confusion Matrix

		Predicted Trajectories	
Actual Trajectories		Normal	Outlier
	Normal	45	4
	Outlier	2	9

Table 3 Confusion Table

TRUE POSITIVE(TP) Normal Trajectories correctly marked as Normal	FALSE NEGATIVE(FN) Normal Trajectories incorrectly marked as Outlier	P
45	4	49
FALSE POSITIVE(FP) Outlier Trajectories incorrectly marked as Normal	TRUE NEGATIVE(TN) Outlier trajectories correctly marked as	N

	Outlier	
2	9	11

Sensitivity, recall, Hit rate or true positive rate(TPR)

$$\mathbf{TPR} = \frac{\mathbf{TP}}{\boldsymbol{P}} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FN}}$$

TPR=45/49=.91836

Specificity or true negative rate (TNR)

$$\mathbf{TNR} = \frac{\mathbf{TN}}{\boldsymbol{N}} = \frac{\mathbf{TN}}{\mathbf{TN} + \mathbf{FP}}$$

TNR= 9/11=.818

Precision or positive predictive value (PPV)

$$\mathbf{PPV} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FP}}$$

PPV=45/45+2=.95744

Negative predictive value (NPV)

$$\mathbf{NPV} = \frac{\mathbf{TN}}{\mathbf{TN} + \mathbf{FN}}$$

NPV=9/9+4=.69230

Miss rate or false negative rate (FNR)

$$\mathbf{FNR} = \frac{\mathbf{FN}}{\boldsymbol{P}} = \frac{\mathbf{FN}}{\mathbf{FN} + \mathbf{TP}} = 1 - \mathbf{TPR}$$

FNR=4/49=.08163

Fall-out or false positive rate (FPR)

$$\mathbf{FPR} = \frac{\mathbf{FP}}{\boldsymbol{N}} = \frac{\mathbf{FP}}{\mathbf{FP} + \mathbf{TN}} = 1 - \mathbf{TNR}$$

FPR=2/11=.1818

False discovery rate (FDR)

$$\mathbf{FDR} = \frac{\mathbf{FP}}{\mathbf{FP} + \mathbf{TP}} = 1 - \mathbf{PPV}$$

FDR=2/2+45=.04255

False omission rate (FOR)

$$\mathbf{FOR} = \frac{\mathbf{FN}}{\mathbf{FN} + \mathbf{TN}} = 1 - \mathbf{NPV}$$

FOR=4/4+9=.30769

Accuracy (ACC)

$$\mathbf{ACC} = \frac{\mathbf{TP} + \mathbf{TN}}{\boldsymbol{P} + \boldsymbol{N}} = \frac{\mathbf{TP} + \mathbf{TN}}{\mathbf{TP} + \mathbf{TN} + \mathbf{FP} + \mathbf{FN}}$$

ACC=45+9/49+11=.9

F1 score is the harmonic mean of precision and sensitivity

$$\boldsymbol{F_1} = 2 \cdot \frac{\mathbf{PPV} \cdot \mathbf{TPR}}{\mathbf{PPV} + \mathbf{TPR}} = \frac{\mathbf{2TP}}{\mathbf{2TP} + \mathbf{FP} + \mathbf{FN}}$$

F_1=2 * 45/(2*45+2+4)=.9375

Markedness (MK)

$$\mathbf{BM} = \mathbf{TPR} + \mathbf{TNR} - 1$$

BM=.91836+.818=.10036

6. CONCLUSION

This paper, presents a novel framework, for trajectory outlier detection based on hausdorff distance. Proposed an algorithm **Out-tra**, for trajectory outlier detection based on hausdorff distance. The visual inspection of results show that Out-tra effectively detects trajectory outliers from a trajectory database. Out-tra measures trajectory dissimilarity using Hausdorff distance. An advantage of **Out-tra** is detection of anomalous behavior of partial section of the trajectory as well. Work has been tested for real dataset, School Buses, for the different values of parameters. Results from experiments show that the proposed method achieved good accuracy with different parameters. In future it can be hybridized with road network, which check the trajectories for anomalous behavior. Combining the work with traffic control will provide solution for dealing with adverse activity like unusual driving, breaking traffic signals etc.

7. REFERENCES

[1] Knorr, E. M., Ng, R. T., & Tucakov, V. (2000). Distance-based outliers: algorithms and applications. *The VLDB Journal—The International Journal on Very Large Data Bases*, *8*(3-4), 237-253.

[2] Giannotti, F., Nanni, M., Pinelli, F., & Pedreschi, D. (2007, August). Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 330-339). ACM.

[3] Aggarwal, C. C., & Yu, P. S. (2001). Outlier detection for high dimensional data. In *ACM Sigmod Record* (Vol. 30, No. 2, pp. 37-46). ACM.

[4] Jin, W., Tung, A. K., & Han, J. (2001). Mining top-n local outliers in large databases. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 293-298). ACM.

[5] Agarwal, P. K., Har-Peled, S., Sharir, M., & Wang, Y. (2003, June). Hausdorff distance under translation for points and balls. In *Proceedings of the nineteenth annual symposium on Computational geometry* (pp. 282-291). ACM.

[6] Cheng, T., & Li, Z. (2004). A hybrid approach to detect spatial-temporal outliers. In *Proceedings of the 12th International Conference on Geoinformatics Geospatial Information Research* (pp. 173-178).

[7] Li, X., Han, J., Kim, S., & Gonzalez, H. (2007, April). Roam: Rule-and motif-based anomaly detection in massive moving object data sets. In *Proceedings of the 2007 SIAM International Conference on Data Mining* (pp. 273-284). Society for Industrial and Applied Mathematics.

[8] Huttenlocher, D. P., Klanderman, G. A., & Rucklidge, W. J. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on pattern analysis and machine intelligence*, *15*(9), 850-863.

[9] Lee, J. G., Han, J., & Li, X. (2008, April). Trajectory outlier detection: A partition-and-detect framework. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on* (pp. 140-149). IEEE.

[10] Ramaswamy, S., Rastogi, R., & Shim, K. (2000, May). Efficient algorithms for mining outliers from large data sets. In *ACM Sigmod Record* (Vol. 29, No. 2, pp. 427-438). ACM.

[11] Pokrajac, D., Lazarevic, A., & Latecki, L. J. (2007, March). Incremental local outlier detection for data streams. In *Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on* (pp. 504-515). IEEE.

[12] Lee, J. G., Han, J., & Whang, K. Y. (2007, June). Trajectory clustering: a partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data* (pp. 593-604). ACM.

[13] Theodoridis, Y. (2003). The R-tree-portal. URL: www. rtreeportal. org (accessed 15 March 2006).

[14] Mirge, V., Gupta, S., & Verma, K. (2014). A Novel Approach for Mining Trajectory patterns of Moving Vehicles. *International Journal of Computer Applications*, *104*(4).

[15] Mirge, V., Verma, K., & Gupta, S. (2016). Dense traffic flow patterns mining in bi-directional road networks using density based trajectory clustering. *Advances in Data Analysis and Classification*, 1-15.