Spatio-Temporal Outlier Detection Technique

K.P. Agrawal, Sanjay Garg andPinkal Patel
Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad, India.
kpa229@gmail.com, gargsv@gmail.com, pinkal08cece@gmail.com

Abstract: Outlier detection is very important functionality of data mining, it has enormous applications. This paper proposes a clustering based approach for outlier detection using spatio-temporal data. It uses three step approach to detect spatio-temporal outliers. In the first step of outlier detection, clustering is performed on the spatio-temporal dataset with proposed Spatio-Temporal Shared Nearest Neighbor (ST-SNN) clustering approach, which is capable to handle high dimensional spatio-temporal data having different sizes and densities and also capable to identify arbitrary shaped cluster. Proposed clustering approach first finds nearest neighbors of each data points and after that it finds the shared nearest neighbor similarity between pair of points in terms of how many nearest neighbors the two points share. Using this similarity measure, our algorithm identifies core points and build clusters around the core points. In the second step of outlier detection, spatial outliers are identified. Finally, in the third step, to find presence of outliers in our dataset, identified spatial outliers are compared with temporal neighbors. The experimental results show that proposed approach is performing much better to identify outliers, especially in high dimensional spatial-temporal data.

Keywords. Outlier Detection, Spatio-Temporal Data, Clustering, Shared-Nearest Neighbors.

1. Introduction

Spatial as well as spatio-temporal databases are growing very quickly, increasing the need for effective and efficient analysis methods to mine the information contained in the data. Most of the Knowledge discovery in databases (KDD) process focuses on the finding hidden patterns in it. However, for various applications like credit card fraudence, discovery of criminal activities in e-commerce, weather prediction etc. require finding outliers (rare events), which are much deviated from remaining observations. Identifying outliers is an important area of research in the field of data mining for variety of applications. Outlier detection has been studied in various data domains which requires dedicated techniques of different types.

Recently, some studies have been proposed on outlier detection [13] [14] [15] on spatial datasets. Most of these studies have been not considered temporal dimension. This paper presents a new clustering based spatio-temporal outlier detection technique, which is based on the shared nearest neighbor concept to cluster the data objects. Clustering algorithms like ROCK [16], DBSCAN [7], and CURE [17] can also handle outliers, but their main concern is to find clusters, and noise points are represented as outliers. Number of noise formed by any clustering algorithm is dependent on a particular algorithm and also on its input parameters.

The rest of the paper is organized as follows. Section 2 describes the related works for outlier detection and problems with them. Section 3 describes basic terminologies of density based clustering approach. Section 4 explains our algorithm to detect outliers inspatio-temporal data. Section 5 presents the experimental results on our NDVI dataset. Finally section 6 presents conclusion and future work.

2. Related Works

This section discusses the existing outlier detection approaches and then shows the various definitions:

2.1. Outlier Detection Approaches

Existing outlier detection approaches can be classified into various categories, these are: distance based, density based, depth based, clustering based, distribution based.

Distance BasedTechnique This method uses a distance metric to measure the distances between the data points. Euclidian distance, Manhattan distance etc. can be used as distance metric [18] [19][23]. Distance based method may create problem if the data parameters are very different from each other in different region of the data set.

Density BasedTechnique This approach was proposed by M. Breunig et al. [13]. This method assign to each object a *degree* of being an outlier. This degree is named as *local outlier factor* (LOF) of an object. Object which having higher LOF value are detected as outliers.

DOI: 10.090592/IJCSC.2015.636

Distribution BasedTechnique These approaches tries to fit some standard statistical distribution model (Normal, Poisson, etc.) to the dataset and identify outliers which deviate to that model [20]..

Depth BasedTechnique These approaches are based on computational geometry, it searches for outliers at the border of the data and independent of statistical distributions. Data objects are organized in convex hull layers [21] [22]. Normal objects are in the center of the data space and Outliers are located at the border of the data space.

Shared Nearest NeighborTechnique This approach is based on the shared nearest neighbor similarity. This approach was first proposed by R.A. JARVIS and EDWARD A. PATRICK [25].In Jarvis-Patrick approach, a shared nearest neighbor (SNN) graph is constructed from the similarity matrix by a link is created between a pair of points, A and B, if and only if A and B have each other in their k-nearest neighbor lists. This process is named as k-nearest neighbor sparsification. The weights of the links between two points in the SNN graph is the number of nearest neighbor two points share.

Cluster BasedTechnique Clustering algorithms can be categorized into five main types: Partitioned, Hierarchical, Model based, Grid and Density based. In our study, we have chosen DBSCAN algorithm, as it is density based clustering algorithm and it is able to discover arbitrary shaped clusters and it does not require the number of clusters as input. Clustering algorithms like CLARANS [1], ROCK [16], DBSCAN [7], can also handle outliers, but their main focus is identify clusters. Here outliers are objects which do not belong to any cluster, also known as noise.

However above mentioned techniques do not consider temporal aspect of data, which is major shortcoming of these techniques. Our proposed algorithm takes into consideration spatial and temporal aspects as well.

3. Basic Concepts

Density based clustering algorithm DBSCAN [7] is used to identify arbitrary shaped clusters and also find noise points in any dataset.ST-DBSCAN[6] is modified DBSCAN to cluster spatio-temporal data.DBSCAN takesEps as one radius value whereas ST-DBSCAN takes Eps1 and Eps2 as radius value for spatial and non-spatial value respectively and a parameter MinPtsused by both algorithms for the number of minimum points that should occur within Eps radius for DBSCAN, and within Eps1 and Eps2 radius for ST-DBSCAN.Basic concepts[6][7]are defined below to explain density based clustering:

(1) Neighbors of an object: It is defined by any distance or similarity measure like Euclidean Distance, Manhattan Distance, and Cosine Similarity etc.for two points p and q, denoted by distance (p,q).

(2) Eps - Neighbors of an object: The Eps-neighbors of an object p is defined by $\{q \epsilon\ D \mid dist\ (p,q) \leq \varepsilon\}$.

(3) Core Object: Point p is called as core object if it contains minimum number of points (MinPts) within radius (Eps).

(4) Directly density-reachable: An object p is directly density-reachable from the object q if p is within the ε -neighbourhood of q, and q is a core object.

(5) Density-reachable:A point p is density-reachable from point q w.r.toEps&MinPts, if there is a chain of points $p_1 \ldots p_n$, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density-reachable from p_i.

(6) Density-connected: Point p and q are density-connected w.r.toEps, MinPts if both the points are density-reachable from a point o w.r.toEps&MinPts.

(7) Cluster: Cluster C w.r.toEps, MinPts is non-empty subset of Dataset D should satisfy the condition of Maximally and Connectivity:

- ∀p,q : if p ϵ C and q is density reachable from p w.r.toEps, MinPts then qϵ C.
- ∀p,qϵ C: p is density-connected to q w.r.toMinPts and MinPts.

4. Proposed Approach

Proposed approach presents new outlier detection technique named as ST-SNN which is based on the Shared Nearest Neighbor Similarity and modified version of existing density based clustering approach. It focuses on the clustering technique to detect outliers, which works well for high dimensional, arbitrary shaped, size and different density dataset. In particular, algorithm first finds k-nearest neighbors for each data objects and then, as in SNN clustering approach, shared nearest neighbors are found between the pairs of points in terms of how many nearest neighbors the two points share. Using ST-SNN clustering, core points are identified in spatio-temporal data and clusters are created around the core points. The use of Shared Nearest Neighbor approach

DOI: 10.090592/IJCSC.2015.636

solves the problems with variable density and the unreliability of distance measure in high dimensions, while the use of core points handles problems with shape and size.

Stepsto be performed in proposed algorithm are: Clustering, Identifying Spatial Outliers, Identifying Temporal Outliers and finally Spatio-temporal outliers are detected.

4.1. Clustering

Clustering is one of the approach to detect outliers. From clustering algorithms point of view, possible outliers are the objects which are not located in any cluster. Additionally, if a cluster is significantly different from other clusters, the objects in this cluster might be possible outliers [9].

A clustering algorithm should be able to discover arbitrary shape clusters of different sizes and densities from high dimensional dataset. Shared Nearest Neighbor (SNN) clustering algorithm [11] is satisfies all these requirements. But it can’t consider temporal aspects when clustering and it can’t detect outliers.In order to overcome these disadvantages and to detect spatio-temporal Outliers, This work improved SNN clustering algorithm to identify spatio-temporal clusters.

4.1.1. Steps to computespatio-temporal shared nearest neighbor (SNN) similarity

Step (1): Compute Similarity Matrix

Compute K - Nearest Neighbors (KNN) (figure 1) for each spatial as well as non-spatial object and find objects which are both spatial neighbors and non-spatial neighbors of an object.

```
For i = 1 to N
        NN (Object$_i$) = KNN (Spatial Object$_i$) ∩ KNN (Non - Spatial Object$_i$)
End For
```

Figure 1. Computation of KNN.

Step (2): Compute Shared Nearest Neighbor Matrix[11]

From the above matrix we can calculate SNN matrix. For each object of the dataset, find the number of neighbors it shares with other objects of the dataset and store in the matrix. Specifically, if A and B are two objects, then SNN similarity between A and B are defined (figure 2) as:

$$Similarity\ (A, B) = length\ (NN\ (A) \cap NN\ (B))$$

```
For i = 1 to N
        For j = 1 to N
            Similarity (i, j) = length (NN (i) ∩ NN (j))
        End For
End For
```

Figure 2. Computation of SNN Matrix.

For all the observations, it creates n X n matrix, where n is the number of observations.ST-SNN algorithm (as shown in figure 3) requires these parameters: K,Eps, Δε1and Δε2. Parameter K is the number of nearest neighbors for each spatial as well as non-spatial data object. Eps is defined as number of points the two point shares.MinPts is defined as the minimum number of points within Eps radius. Parameters Eps and MinPts are taken as fraction of K. Here, K, Eps and MinPts are identified by trial and error method. In ST-SNN algorithm we are using two parameters Δε1 and Δε2 to avoid the determining of combined clusters if there is slight differences in the values of neighbor locations.Δε1 is used for spatial values whereas Δε2 is used for non-spatial values.Δε1 and Δε2 are identified through K-Distance Heuristic suggested in [7]. To determine Δε1 and Δε2 the first step of heuristic is to find k-nearest neighbors distance for each object, where k is equal to MinPts. Then sort these k-distance values in descending order. The first “valley” of the sorted graph is the threshold point.

```
ST_SNN_Clustering (Dataset, K, Δε1, Δε2)
Eps = fraction of K
MinPts = fraction of K
        For i in 1 to N
                If Obj$_I$is not belongs to any cluster, then
X = FIND_SNN (Obj$_I$, Eps)
If |X| <MinPts Then
          Mark Obj$_I$as noise.
Else
Cluster_Label = Cluster_Label + 1
                      For j = 1 to |X|
                               Mark all objects in X with current Cluster_Label
                      End For

                      For k=1 to |X|
                               Y = FIND_SNN (Obj$_k$, Eps)

                                   If |Y| >= MinPts Then
                                          For All objects obj in Y
If (obj is not marked as noise OR it is not in a cluster) AND
                                                 (|Cluster_Avg () – obj.Spatial_Value|<=Δε1) AND
                                                 (|Cluster_Avg () – obj.Non_Spatial_Value|<=Δε2) Then
                                                            Mark Obj with current Cluster_Label.
                                               End If
                                          End For
                                   End If
                    End For
                End If
            End If
      End For
```

Figure 3.ST_SNN_Clustering.

4.2 Detection of ST-Outliers

In order to identify outliers in our spatio-temporal data we need to identify spatial outliers and then compare the spatial outlier regions with other objects of same local area but at different times. If the values of spatial outlier regions are not different with its temporal neighbors with appropriate measure then that region is not reported as outlier, otherwise that region is reported as outlier region.

4.2.1Identifying Spatial Outliers

In the previous step,clusters of arbitrary shaped, variable densities are identified and noise points are also detected. Outliers for NDVI datasets are those good vegetation regions where vegetation occurred very low for a particular year or those bad vegetation regions where vegetation occurred very high for a particular year. So, that region is detected as spatial outlier region.

On that basis,if there exist an outlier region than that outlier region for a particular year will merge either with low vegetation or with high vegetation area in clustering phase. It needs to extract that outlier region if it exist from clusters. For that purpose it first identify the low vegetation and high vegetation regions. This can be done by taking average for each cluster identified in clustering phase, and arrange them in ascending order. In this ordering step top-k low vegetation regions on the top of the list andtop-k high vegetation regions on the bottom of the list are extracted, where k can be either 1, 2 or 3. Because higher k - valued regions will never contain outliers. These top-k low vegetation and top-k high vegetation regions are reported as spatial outliers regions.

4.2.2Identifying Temporal Outliers

This step will extract the outlier regions from the detected spatial outlier region in previous step. Two objects are temporal neighbors if the values of these objects are observed in consecutive time units [9]. Spatial outlier regions are compared with temporal neighbors and if the values of spatial outlier regions are different with its temporal neighbors then that region is reported as outlier.

5. Experimentation

We have a spatio-temporal dataset which contains NDVI values for different states in India. These NDVI values shows low vegetation when values are between 0.1 and 0.3 and shows high vegetation when values are between 0.8 and 1. Our Dataset has the columns: Grid code for each grid, Latitude of the grid, Longitude of the grid, state, City and 23 columns for NDVI values, where each column contains 16 days composite NDVI values.

5.1 Implementation Details and Results

During implementation phase, we first cluster our dataset to find the regions where vegetation characteristics are similar. We have performed our experiment on Gujarat region. We have selected data set for the year 2003. The input parameters chosen as k=200, Eps=4 (2% of k), MinPts=6 (3% of k), Δε1=0.2022, Δε2=0.5861. Where k, Eps, MinPts are selected by trial and error method and Δε1, Δε2 are identified through k-distance graph. We performed our experimentation on k= (50, 100, 200) and selected best k value.
Using these parameters we identified clusters shown in figure5.

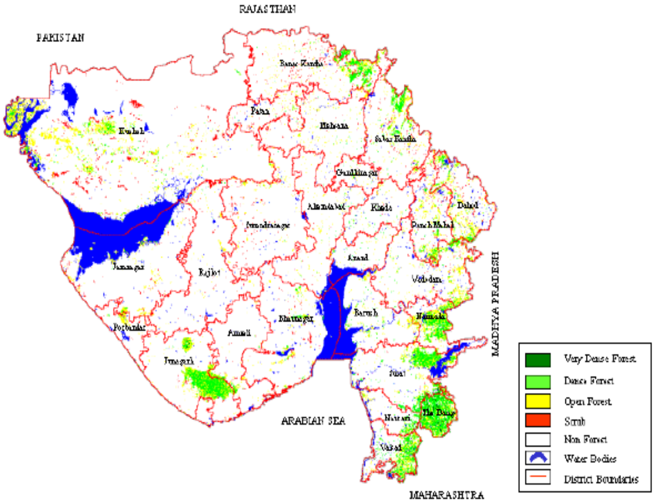


Figure 4. Gujarat region's forest survey map.

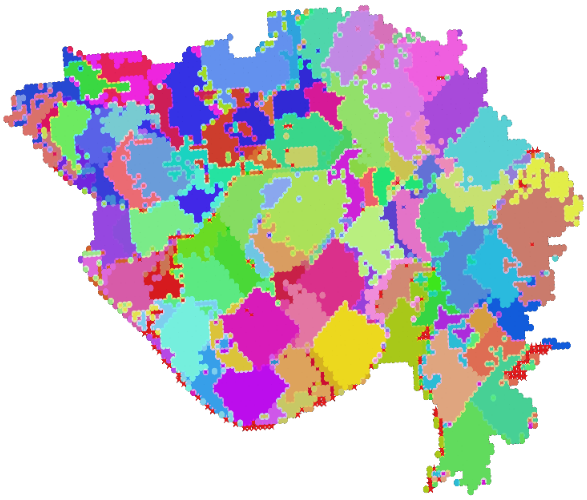


Figure 5. Clusters identified by ST_SNN_Clustering

Figure 5 shows the clustering results by our algorithm and figure 4 is the Gujarat map taken from forest survey of India [25]. Figure 4 is added for validation purpose to check that our clustering algorithm is giving good cluster formation or not but comparing both figure we can't say anything, because number of clusters reported by our proposed algorithm are more. To make our results comparable, agglomerative clustering is used to merge the clusters. Agglomerative clustering required number of clusters as input parameter, so for this purpose we have taken crop list from Gujarat agriculture department [24], where number of crops are 20, so 20 as the number of clusters are given input to the agglomerative clustering.

DOI: 10.090592/IJCSC.2015.636

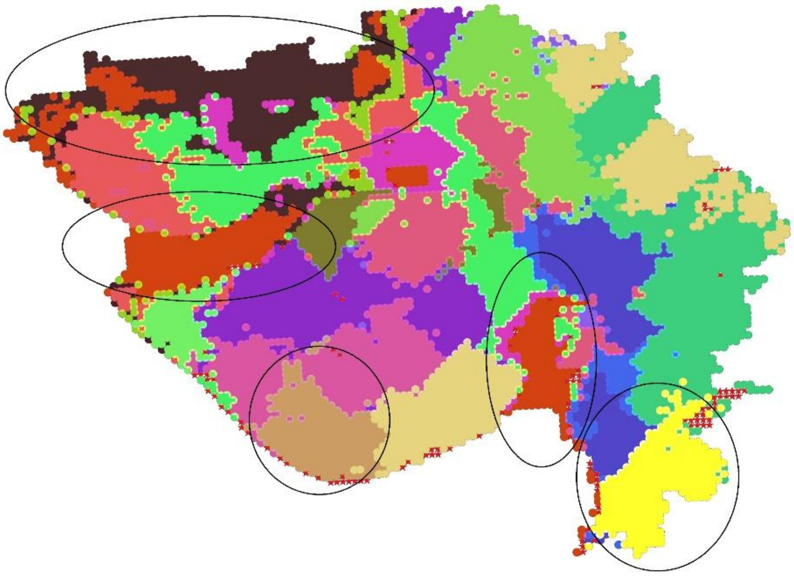


Figure 6. Clusters after merging.

Figure 6 shows clustering result after merging. Encircled regions are compared and it shows that our spatio-temporal shared nearest neighbor clustering algorithm forms good quality clusters. Then identification of spatial outliers in this dataset where vegetation are very low for that particular year and spatial outlier region are shown in figure7.

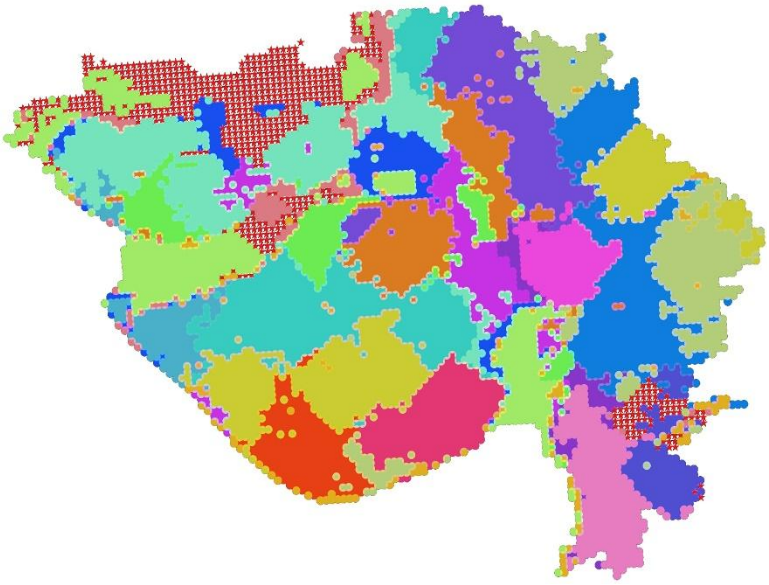
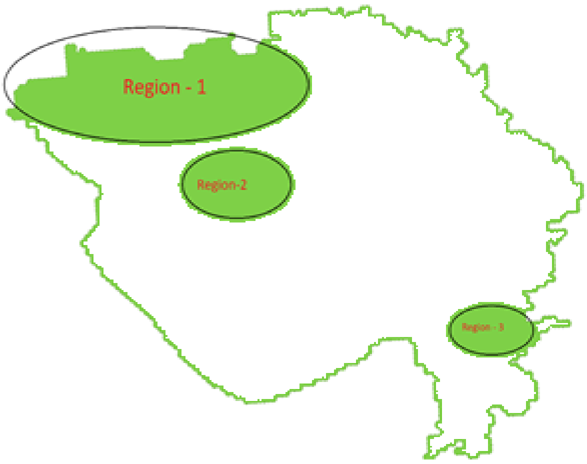


Figure 7. The region marked as star shows spatial outliers.

The region star marked in figure 7 has significantly low vegetation values on 2003. This region has vegetation values approximately in the range of 0.05 to 0.2, so it contains spatial outliers. Next it checks the temporal neighbors. The vegetation values of spatial outliers with other data points of the same location but in different time units are compared. So, spatial outliers'locations of year 2003 with the years 2002 and 2004 are compared. Figure 8 shows the potential spatial outlier regions (Region-1, Region-2 and Region-3) in the year 2002 and 2004. By comparison it identifiesthat Region-1 and Region-2 of the spatial outlier region is having same vegetation values in the years 2002 and 2004 but Region-3 having significantly different vegetation values.

DOI: 10.090592/IJCSC.2015.636



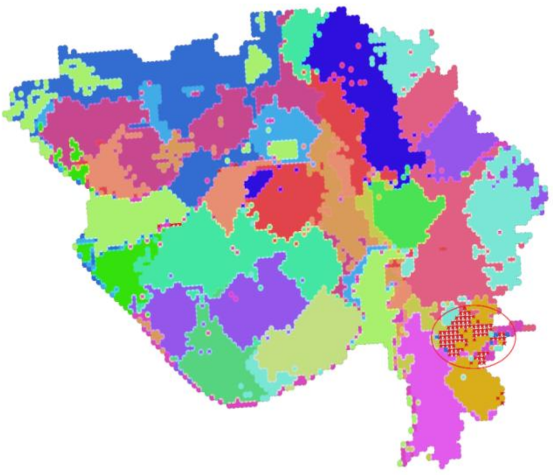


Figure 8. Potentialspatial outlier region in year 2002 and 2004.**Figure 9.**Spatio-Temporal Outlier Region.

Authors have detected that the region encircled in figure 9 have too low vegetation values. For this region, the objects in this regions are confirmed as Spatio-Temporal Outlier.

The average runtime complexity of our outlier detection algorithm is O (n^2), where n is the number of objects in the dataset. The space complexity of the algorithm is also O (n^2), since we need to store k-nearest neighbors which takes O (k*n) space and shared nearest neighbors similarity matrix which takes O (n^2). While the k-nearest neighbors and shared nearest neighbors similarity matrix can be computed once and used repeatedly for different parameter values of the algorithm.

6. Conclusions and future scope

This paper proposed a spatio-temporal outlier detection approach in large high dimensional spatio-temporal datasets. To detect outliers in spatio-temporal dataset, a new spatio-temporal clustering approach is used which is based on shared nearest neighbor concept. This clustering algorithm can find clusters of varying shapes, sizes, densities and automatically identifies the number of clusters. Agglomerative clustering approach has been used to merge theobtained clusters tointerpret and comparewith forest survey of India map. Then we identified spatial outliers and then finally spatio-temporal outliers are detected in NDVI dataset which have been shown in experimental results. Experimental results demonstrate that our outlier detection approach is very promising for spatio-temporal data.

For future work, improvement of the run time complexity can be done by parallelizing the algorithm. In addition, heuristic can be improvised to determine the input parameters K, Eps and MinPts.

Acknowledgement
This work is financially supported by Space Application Centre – Indian Space Research Organization (SAC-ISRO) under RESPOND Scheme vide NO. ISRO/RES/4/608/2013-14 dated on 22.4.13. Authors would like to extend their sincere thanks to SAC-ISRO authorities.

References

[1] Raymond T. Ng and Jiawei Han, CLARANS: A Method for Clustering Objects for Spatial Data Mining 2002.

[2]M. Gupta, J.Gao, C. Aggarwal, and J. Han, “Outlier detection for temporal data: A survey," Knowledge and Data Engineering, IEEE Transactions on, vol. 26, pp. 22502267, Sept 2014.

[3] V. Hodge and J. Austin, “A survey of outlier detection methodologies," Artif. Intell.Rev., vol. 22, pp. 85-126, Oct. 2004.

[4] S.-Y. Jiang and Q. bo An, “Clustering-based outlier detection method," in FuzzySystems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conferenceon, vol. 2, pp. 429-433, Oct 2008.

[5]V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey," ACMComputerSurv. vol. 41, pp. 15:1-15:58, jul 2009.

[6]D. Birant and A. Kut, “St-dbscan: An algorithm for clustering spatial-temporaldata," Data Knowl. Eng., vol. 60, pp. 208-221, Jan. 2007.

[7]M. Ester, H. peter Kriegel, J. S, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise," AAAI Press, pp. 226-231, 1996.

[8]M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, “Optics: Ordering pointsto identify the clustering structure," SIGMOD Rec., vol. 28, pp. 49-60, June 1999.

[9]D. Birant and A. Kut, “Spatio-temporal outlier detection in large databases," inInformation Technology Interfaces, 2006. 28th International Conference on, pp. 179-184, 2006.

[10]R. Jarvis and E. A. Patrick, “Clustering using a similarity measure based on sharednear neighbour’s," Computers, IEEE Transactions on, vol. C-22, pp. 1025-1034, Nov1973.

[11]L. Ertz, M. Steinbach, and V. Kumar, “Finding clusters of different sizes, shapes,and densities in noisy, high dimensional data," in In: Proceedings of Second SIAMInternational Conference on Data Mining, 2003.

[12]L. Ertoz, M. Steinbach, and V. Kumar, “A new shared nearest neighbour clusteringalgorithm and its applications," in Workshop on Clustering High Dimensional Dataand its Applications at 2nd SIAM International Conference on Data Mining, 2002.

[13]Breunig MM, Kriegel H-P, Ng R, Sander J. LOF: Identifying Density-Based Local Outliers, ACM SIGMOD Int. Conf. on Management of Data, Dallas, p. 93-104, 2000.

[14]Papadimitriou S, Faloutsos C. Cross-Outlier Detection, In: Proc. 8th International Symposium on Spatial and Temporal Databases, Greece; 199-213, 2003.

[15]Zengyou He, XiaofeiXu, Shengchun Deng, “Discovering cluster-based local outliers”, in Pattern Recognition Letters; 9-10, 2003

[16]Guha, S., Rastogi, R., Kyuseok, S., ROCK: A robust clustering algorithm for categorical attributes. In: Proceedings of ICDE_99, Sydney, Australia, pp. 512–521, 1999.

[17]Guha, Sudipto; Rastogi, Rajeev; Shim, Kyuseok, CURE: An Efficient Clustering Algorithm for Large Databases. In: Proceedings of Information Systems 26, 35-58, 2001.

[18]Knorr EM, Ng RT. Algorithms for Mining Distance-Based Outliers in Large Datasets, In: Proc. 24th Int. Conf. Very Large Data Bases, New York, NY; p. 392-403, 1998.

[19]Knorr EM, Ng RT, Tucakov V. Distance Based Outliers: Algorithms and Applications, Journal: Very Large Data Bases 8 (3-4): 237-253, 2000.

[20]Barnett V, Lewis T. Outliers in Statistical Data. New York: John Wiley; 1994.

[21] Johnson T, Kwok I, Ng R. Fast Computation of 2-Dimensional Depth Contours, In: Proc. 4th. Int. Conf. on KDD, New York, NY; p. 224-228, 1998.

[22] Ruts I, Rousseeuw P. Computing Depth Contours of Bivariate Point Clouds, Journal of Computational Statistics and Data Analysis,153-168, 1996.

[23] S. Garg and R. C. Jain, Variations of k-mean Algorithm: A Study for High-Dimensional Large Data Sets, Information Technology Journal 5(6), pp. 1132-1135, 2006.

[24] State of Forest Report, Forest Survey of India,Ministry of Environment & Forest, Government of India, Downloaded on February 2015,http://fsi.nic.in/details.php?pgID=qu_4.

[25] State of Gujarat Agriculture Department,Downloaded on January 2015, http://www.agri.gujarat.gov.in/.