*IOSR Journal of Computer Engineering (IOSRJCE)*
*ISSN: 2278-0661 Volume 3, Issue 6 (Sep-Oct. 2012), PP 01-09*
*www.iosrjournals.org*

# A Density Based Clustering Technique For Large Spatial Data Using Polygon Approach

Hrishav Bakul Barua[1], Dhiraj Kumar Das [2] & Sauravjyoti Sarmah[3]
[1, 2](Dept. of Information Technology, (Student) Sikkim Manipal Institute of Technology/ Sikkim ManipalUniversity, India)
[3](Dept. of Computer Science and Engineering, (Faculty) Jorhat Engineering College/Dibrugarh University, India)

***Abstract :*** *Finding meaningful patterns and useful trends in large datasets has attracted considerable interest recently, and one of the most widely studied problems in this area is the identification and formation of clusters, or densely populated regions in a dataset. Prior work does not adequately address the problem of large datasets and minimization of I/O costs. The objective of this paper is to present a Triangle-density based clustering technique, which we have named as TDCT, for efficient clustering of spatial data. This algorithm is capable of identifying embedded clusters of arbitrary shapes as well as multi-density clusters over large spatial datasets. The Polygon approach is being used to perform the clustering where the number of points inside a triangle (triangle density) of a polygon is calculated using barycentric formulae. This is because of the fact that partitioning of the data set can be performed more efficiently in triangular shape than in any other polygonal shape due to its smaller space dimension. The ratio of number of points between two triangles can be found out which forms the basis of nested clustering. Experimental results are reported to establish the superiority of the technique in terms of cluster quality and complexity.*

***Keywords:*** *Clustering, Density- based, Density Confidence, Polygon approach, Triangle-density.*

## I. INTRODUCTION

In this paper, the technique of data clustering has been examined, which is a particular kind of data mining problem. The process of grouping a set of physical or abstract objects into classes of similar objects is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters [1]. Given a large set of data points (data objects); the data space is usually not uniformly occupied. Data Clustering identifies the sparse and the crowded places, and hence discovers the overall distribution patterns of the data set. Besides, the derived clusters can be visualized more efficiently and effectively than the original dataset. Mining knowledge from large amounts of spatial data is known as spatial data mining. It becomes a highly demanding field because huge amounts of spatial data have been collected in various applications ranging from geo-spatial data to bio-medical knowledge. The amount of spatial data being collected is increasing exponentially and has far exceeded human's ability to analyze them. Recently, clustering has been recognized as a primary data mining method for knowledge discovery in spatial database. The development of clustering algorithms has received a lot of attention in the last few years and new clustering algorithms are proposed.

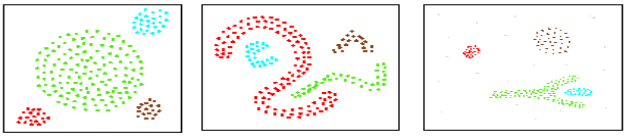

Figure 1- Formation of clusters

Major clustering techniques have been classified into partitional, hierarchical, density-based, grid-based and model-based. Among these techniques, the density-based approach is famous for its capability of discovering arbitrary shaped clusters of good quality even in noisy datasets [2]. Density Based Spatial Clustering of Applications with Noise (DBSCAN) and Ordering Points to Identify the Clustering Structure (OPTICS) are two of the most popular density based clustering algorithms. In density-based clustering algorithms, a cluster is defined as a high-density region partitioned by low-density regions in data space. They can find out the clusters of different shapes and sizes from the large amount of data containing noise and outliers. Fig. 1 depicts the formation of clusters of similar data. This project, which is primarily motivated by Density Based Clustering, aims at proposing a new density based clustering technique for efficient clustering of spatial data. The rest of the paper is organized as follows. Section 2 provides a selected literary review on density based, grid based and other multi-density as well as variable density data clustering techniques. Section 3 illustrates the background of the proposed work and section 4 gives the final proposed algorithm. In section 5, we present the experimental results and the performance analysis of the work following the complexity analysis in section 6. Lastly, we

conclude with a summary and conclusion in section 7. Section 8 acknowledges the organizations that guided us through this research.

## II. RELATED WORKS

This section portrays a selected literary review on some of the previous works in this field specially some relevant density based as well as grid based clustering techniques.

***A. Density Based Approach:*** Most partitioning methods cluster objects based on the distance between the objects. Such methods can find only spherical-shaped clusters and encounter difficulty at discovering clusters of arbitrary shapes. Other clustering methods have been developed based on the notion of the density. Their general idea is to continue growing the given cluster as long as the density (number of objects or data points) in the neighborhood exceeds some threshold; that is for each data point within a given cluster, the neighborhood of a given radius has to contain at least a minimum number of points. Such a method can be used to filter out noise (outliers) and discover clusters of arbitrary shape. To discover clusters with arbitrary shape, density based clustering methods have been developed. These typically regard clusters as dense regions of objects in the data space that are separated by regions of low density. DBSCAN [2] grow clusters according to a density-based connectivity analysis. OPTICS extends DBSCAN [2] to produce a cluster ordering obtained from a wide range of parameter settings. The idea behind density based clustering approach is that the density of points within a cluster is higher as compared to those outside of it. DBSCAN [2] is a density based clustering algorithm capable of discovering clusters of various shapes even in presence of noise. The key idea of DBSCAN is that for each point of a cluster, the neighborhood of a given radius (ε) has to contain at least a minimum number of points and the density in the neighborhood has to exceed some threshold. It is efficient for large spatial databases but, for massive datasets, it becomes very time consuming, even if the use of R* tree is made. Another drawback of DBSCAN is that due to the use of the global density parameters, it fails to detect embedded or nested clusters.

***B. Grid Based Approach:*** Grid-based methods quantize the object space into a finite number of cells that form a grid structure. All of the clustering operations are performed on the grid structure. The main advantage of this approach is its fast-processing time, which is independent of the number of data objects and dependent only on the number of cells in each dimension in the quantized space. There is high probability that all data points that fall into the same grid cell belong to the same cluster. Therefore all data points belonging to the same cell can be aggregated and treated as one object [3]. It is due to this nature that grid-based clustering algorithms are computationally efficient which depends on the number of cells in each dimension in the quantized space. It has many advantages such as the total number of the grid cells is independent of the number of data points and is insensitive ofthe order of input data points. Some of the popular grid-based clustering techniques are STING [4], WaveCluster [5],CLIQUE [6], pMAFIA [7] etc. STING [4] uses a multi resolution approach to perform cluster analysis. Theadvantage of STING is that it is query-independent and easyto parallelize. However the shapes of clusters have horizontalor vertical boundaries but no diagonal boundary is detected. WaveCluster [5] also uses a multidimensional grid structure. Ithelps in detecting clusters of data at varying levels of accuracy. It automatically removes outliers and is very fast. However, it is not suitable for high dimensional data sets.CLIQUE [6] is a hybrid clustering method that combines theidea of both density-based and grid-based approaches. Itautomatically finds subspaces of the highest dimensionality and is insensitive to the order of input.Moreover, it has goodscalability as the number of dimensions in the data increases.However, the accuracy of the clustering result may bedegraded at the expense of simplicity of the method. pMAFIA[7] is an optimized and improved version of CLIQUE. It usesthe concept of adaptive grids for detecting the clusters. Itscales exponentially to the dimension of the cluster of the highest dimension in the data set.

***C. Clustering Over Multi Density Data Space:*** One of the main applications of clustering spatial databases is to find clusters of spatial objects which are close to each other. Most traditional clustering algorithms try to discover clusters of arbitrary densities, shapes and sizes. Very few clustering algorithms show preferable efficiency when clustering multi-density datasets. This is also because small clusters with small number of points in a local area are possible to be missed by a global density threshold. Some clustering algorithms that can cluster on multi-density datasets are Chameleon [8], SNN [9] (shared nearest neighbor), and the multi-stage density-isoline algorithm and so on. Chameleon [8] can handle multi-density datasets, but for large datasets the time complexity is too high. SNN [9] algorithm can find clusters of varying shapes, sizes and densities and can also handle multi-density dataset. The disadvantage of SNN is that the degree of precision is low on the multi-density clustering and finding outliers. The multi-stage density-isoline algorithm [10] clusters datasets by the multi-stage way and the idea of density-isoline. The disadvantage of the algorithm is that each cluster cannot be separated efficiently. DGCL [11] is based on density-grid based clustering approach. But, since it uses a uniform density threshold it causes the low density clusters to be lost.

***D. Clustering Over Variable Density Space:*** Most of the real life datasets have a skewed distribution and may also contain nested cluster structures the discovery of which is very difficult. Therefore, we discuss two density based approaches, OPTICS [12] and EnDBSCAN [13], which attempt to handle the datasets with variable density successfully. OPTICS can identify embedded clusters over varying density space. However, its

execution time performance degrades in case of large datasets with variable density space and it cannot detect nested cluster structures successfully over massive datasets. In EnDBSCAN [13], an attempt is made to detect embedded or nested clusters using an integrated approach. Based on our experimental analysis in light of very large synthetic datasets, it has been observed that EnDBSCAN can detect embedded clusters; however, with the increase in the volume of data, the performance of it also degrades. EnDBSCAN is highly sensitive to the parameters MinPts and ε. In addition to the above mentioned parameters, OPTICS requires an additional parameter i.e. ε'

***Based on our selected survey and experimental analysis, it has been observed that:***

1) Density based approach is most suitable for quality cluster detection over massive datasets.
2) Grid based approach is suitable for fast processing of large datasets.
3) Almost all clustering algorithms require input parameters, accurate determination of which are very difficult, especially for real world data sets containing high dimensional objects. Moreover, the algorithms are highly sensitive to those parameters.
4) None of the techniques discussed above, is capable in handling multi-density datasets as well as multiple intrinsic or nested clusters over massive datasets qualitatively.

***E. Motivation:*** Most partitioning methods cluster objects based on the distance between them. Such methods can find only spherical-shaped clusters and encounter difficulty at discovering clusters of arbitrary shapes. So other clustering methods have been developed based on the notion of density. The general idea is to continue growing the given cluster as long as the density (number of objects or data points) in the "neighborhood" exceeds some threshold; this is, for each data point within a given cluster, the neighborhood of a given radius has to contain at least a minimum number of points. Such a method can be used to filter out noise (outliers) and discover cluster of arbitrary shapes. These typically regard clusters as dense regions of objects in the data space that are separated by regions of low density (representing noise). DBSCAN grow regions with sufficiently high density into clusters and discover clusters of arbitrary shape in spatial database with noise according to a density-based connectivity analysis. It defines a cluster as a maximal set of density-connected points.

The major drawback in such clustering approaches is the processing time. The time to scan the whole database and cluster accordingly is a major area of concern. This project is motivated by the density-based clustering approaches to discover nested clusters and clusters of arbitrary shapes. Fig. 2 displays a Nested cluster and a Multi-density cluster. The basic idea is to reduce the processing time and enhance the efficiency henceforth. For the fulfillment of the requirements of the undertaken research the sampling technique is proposed. It is used as a data reduction technique because it allows a large data set to be represented by a much smaller random sample (or subset) of the data. An advantage of sampling for data reduction is that the cost of obtaining a sample is proportional to the size of the sample, as opposed to the whole data set size. Other data reduction techniques can require at least one complete pass through. Sampling is a natural choice for progressive refinement of a reduced data set. Such a set can be further refined by simply increasing the sample size. This technique of clustering can be efficiently used in a spatial database, where we can choose to define clusters geographically based on how closely different areas are located. Better cluster quality and more acceptable complexity are the major features of the proposed approach.

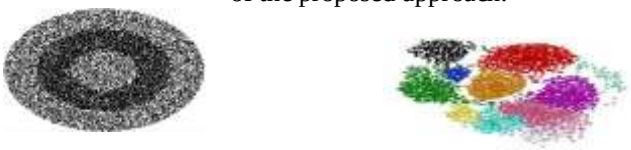

Figure 2- Nested clusters, Multi-density Clusters

## III. THEORITICAL BACKGROUND OF THE PROPOSED WORK

The distribution of data in a data set is not uniform in general. Some portions of the data space are highly dense while some portions are sparse. The problem associated here is efficient clustering of spatial data points to discover nested clusters and clusters of arbitrary shapes using density based clustering technique.

**A. *Definitions:***

**Here, we introduce some definitions which are used in the proposed algorithm:**

*Definition 1* Triangle Density: The number of spatial point objects within a particular triangle of a particular polygon (octagon).

*Definition 2* Useful Triangle: Only those triangles which are populated i.e., which contain data points will be treated as useful triangle.

*Definition 3* Neighbor Triangle: Those triangles which have a common edge (edge neighbors) to the current triangle are the neighbors of the current triangle. Fig. 3 shows the neighbor triangles (in grey shade) of the current triangle 1.

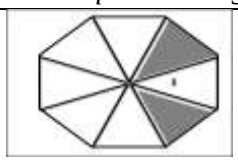

Figure. 3- Neighbor triangles (shaded in gray) of the triangle 1

*Definition 4* Density Confidence of a triangle: If the ratio of the densities of the current triangle and one of its neighbors is greater than β (user's input) then the two triangles can be merged into the same cluster. Therefore the following condition should be satisfied: $\beta \leq d_n (T_{P1}) / d_n (T_{p2})$ where $d_n$ represents the density of the particular triangle.

*Definition 5* Reachability of a triangle: A triangle p is reachable from a triangle q if p is a neighbor triangle of q and triangle p satisfies the density confidence condition w.r.t. triangle q.

*Definition 6* Neighbor Polygon: Those polygons (in our case octagons) which are formed from classified data points (points which lie in merged triangles) of the current polygon as its center are called as neighbor polygons of the current polygon. Fig. 4 shows a neighbor polygon 2 of polygon 1.

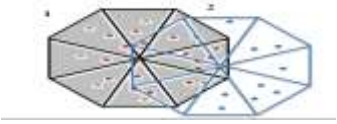

Figure 4- Neighbor polygon 2 of polygon 1

*Definition 7* Reachability of a polygon: A polygon 2 is reachable from polygon 1 if 2 is neighbor polygon of 1.

*Definition 8* Cluster: A cluster is defined to be the set of points belonging to the set of reachable triangles and polygons. A cluster C is a non-empty subset satisfying the following conditions,

For all u, v: if u belongs to C and v is reachable from u then v belongs to C where u and v are polygons (octagon) and,

For each u, v belonging to C, for all p and q that belongs to u or v and p belongs to C and q is reachable from p w.r.t. $\beta$ then q also belongs to C, where p,q are triangles of an octagon . Triangle-reachable relation follows symmetric and transitive property within a octagon of an cluster C.

*Definition 9* Noise: Noise is simply the set of points belonging to the octagons (or triangles) not belonging to any of its clusters. Let $C_1$, $C_2$, ....$C_3$ be the clusters w.r.t. β, then noise = {no_ p| p belongs to dataset, for all i no_ p does not belong to C where no_ p is the set of points in triangle p or polygon u and $C_i$ (i=1,..., k).

**B.** ***Density Confidence:*** The density confidence for a given set of triangles reflects the general trend and distribution of the data points of that set. If the density of one triangle varies greatly from the others it will not be included in the set. If the density confidence of a current triangle with one of its neighbor triangle does not satisfy the density confidence condition than that neighbor triangle is not included into the currently considered dense area. On the contrary, if it satisfies the condition than we treat the neighbor triangle as a part of the considered dense area and merge the triangle with the dense area. In comparison to other methods of setting a global threshold, this method has the ability to recognize the local dense areas in the data space where multi-density clusters exist.

With correspondence to the above definitions, three lemmas are stated:

***Lemma 1***: Let *C* be a cluster w.r.t. $\beta$ and let *u* be any polygon in *C and p be any triangle in u.* Then *C* can be defined as a set, S= {*s* and $s_t$ | *s* is polygon-reachable from *u* and $s_t$ is triangle-reachable from *p* w.r.t. $\beta$ *within every u of C*}

*Proof:* Suppose *T* is a triangle, where *T* belongs to $s_t$ and *T* is not triangle-reachable from *p* w.r.t. $\beta$ or *G* is an octagon, where *G* belongs to *s* and *G* is not polygon-reachable from *u*. But, a cluster according to Def. 8 will be the set of points which are triangle-reachable from *p* and polygon-reachable from *u.* Therefore, we come to a contradiction and hence the proof.

***Lemma 2***: A triangle of a polygon or a polygon corresponding to noise points is not triangle-reachable or polygon-reachable from any of the clusters respectively. For a triangle *p* we have, all *p*: *p* is not reachable from any triangle in *C* i.e. *p* does not belong to *C* and for a polygon *u* we have, all *u: u* is not reachable from any polygon in *C* i.e. *u* does not belong to *C.*

*Proof*: Suppose, *C* be a cluster w.r.t. $\beta$ and let *p* be a triangle of an octagon corresponding to noise points and u be any polygon corresponding to noise points. Let *p* be triangle-reachable from *C*, then *p* belongs to *C*. and *u* be polygon-reachable from *C*, then u belongs to *C*. But, this violates the Def. 9 that noise points are belonging to triangles or octagons that are not triangle-reachable or polygon-reachable from any of the clusters respectively. Therefore, we come to the conclusion that *p* is not reachable from any triangle in *C* and *u* is not reachable from any octagon in *C*.

***Lemma 3***: A triangle *T* or a polygon *G* can be triangle-reachable and polygon-reachable from only a single unique cluster.

*Proof:* Let $C_1$ and $C_2$ are two clusters w.r.t. $\beta$ and let *p* be any triangle and u be any polygon in $C_1$ and *q* is any triangle and v is any polygon in $C_2$. Suppose a triangle *T* is triangle-reachable from both *p* and *q* or a polygon *G* is

polygon-reachable from both *u* and *v*, then *T* belongs to $C_1$ and *T* belongs to $C_2$ or *G* belongs to $C_1$ and *G* belongs to $C_2$ . This will mean that the clusters $C_1$ and $C_2$ should be merged. This violates the basic notion that clusters are unique sets. Thus, we can conclude that if *T* is triangle-reachable from *p* w.r.t. $\beta$, *T* is not triangle-reachable from *q* w.r.t. $\beta$ and if *G* is polygon-reachable from *u*, then *G* is not polygon-reachable from *v*, *i.e*. *T* belongs to *C*1 and *T* does not belong to *C*2 or G belongs to $C_1$ and G does not belong to $C_2$. Therefore the lemma has been proved.

## IV. THE PROPOSED ALGORITHM

There are functional components like creating the polygon, finding the number of points in each triangle of a polygon (density), identify the maximum dense triangle, traversal of neighboring triangles, merging two triangles and finding the farthest point of each merged triangle that are needed to be implemented to give out quality clusters .

**A. *Creating A Polygon*:** Since we are using the polygon approach, we need a data point as a center of the polygon. Any arbitrary unclassified point can be used to start with. Here, we are using an octagon to serve our purpose. Thus we have eight equal triangles after sub-division of the regular octagon. By increasing the number of sides of the polygon from eight to higher we can have higher number of triangular sub-divisions that can lead to more accurate clustering. The octagon is formed by creating eight equal triangles from a point and an initial radius. This initial radius is called the Epsilon Distance, ε given as an input parameter by the user. Fig. 5 shows the octagon formed with ε as radius and any arbitrary unclassified point as center. All the vertices of the triangles are calculated at a ε-distance from the center point using trigonometric formulas. The found vertices are joined from the center using straight lines. Lastly the vertices are joined with one another using straight lines and this goes on iteratively till all the triangles are formed and joined to form the final octagon.

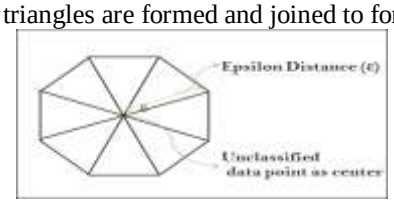


Figure 5-Polygon (octagon) created with neighborhood radius as ε

***B. Compute Density Of Each Triangle*:** The total number of points in each triangle is its density (triangle density). To compute the density we need to find the location of a point wrt the triangles. It is required to identify the points lying inside a triangle. For this purpose the barycentric formulae [17] are used. Using these formulae we find the number of points lying in each triangle and hence find the density of each triangle. Each triangle is given an identifier in numbers (1, 2, 3….., 8). Points are also given identifiers corresponding to which triangle they lie in. Only useful triangles are used for merging. Fig. 6 given below shows the above stated case.

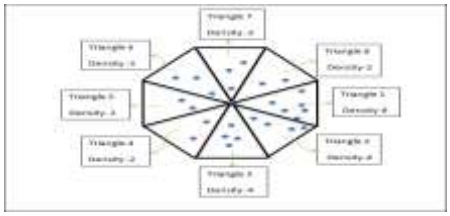

Figure 6- Density of each triangle of the octagon

***C. Identify The Maximum Dense Triangle*:** We will require the densest triangle from the octagon to start with. This is because of the fact that consideration of any other dense triangle other than the maximum dense triangle will result in merging of two clusters with different density. This is obtained by counting the number of points in each triangle and comparing them with each other. This gives out the densest triangle. This is required to start the traversal of neighboring triangles. Fig. 7 portrays this case.

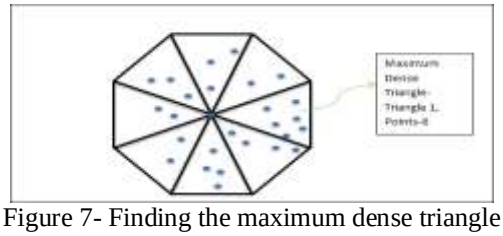

Figure 7- Finding the maximum dense triangle

***D. Traversal Of Neighboring Triangles*:** Starting from the densest triangle we traverse the neighboring triangles in both directions. First we can start in clockwise direction and when completed we may go in anti-

clockwise direction. The basic purpose of traversing is to find the ratio between two adjacent triangles which share the same edge among them (neighbor triangles). Fig. 8 given below shows the possible ways of traversal.

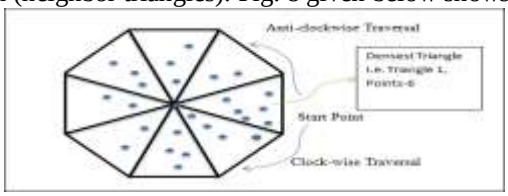


Figure 8- Traversal of neighbor triangles from maximum dense triangle

***E. Merging Two Triangles:*** A triangle can be merged with its neighbor triangle if the ratio of densities between them is greater than a threshold set by the user i.e. the triangle satisfies the density confidence condition w.r.t its neighbor. Thus these reachable triangles can be considered as the same cluster. This process is performed iteratively till no more triangles of an octagon can be merged in either direction. Fig. 9 shows a case in which two neighboring triangles are merged into one cluster satisfying the density confidence condition.

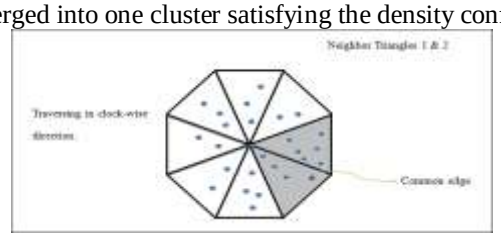


Figure 9- Merging of triangles (in light grey) if density confidence is fulfilled

However, two cases can be considered with regard to density confidence of a triangle

*Case* 1*:* If $T_{P1}$ & $T_{P2}$ are two triangles of a polygon such that $d_n(T_{P1}) > d_n(T_{P2})$, where $d_n$ represents the density of a particular triangle, then : $\beta \geq d_n(T_{P1}) / d_n(T_{p2})$ and $\beta$ is set any value in the interval [2.5,1.43] which is found after rigorous experimentation.

*Case* 2: If $T_{P1}$ & $T_{P2}$ are two triangles of a polygon such that $d_n(T_{P1}) < d_n(T_{P2})$, where $d_n$ represents the density of a particular triangle, then : $\beta \leq d_n(T_{P1}) / d_n(T_{p2})$ and $\beta$ is set any value in the interval [0.4,0.7] which is found after rigorous experimentation.

***F. Finding The Farthest Points From Each Merged Triangle*:** Now we are to proceed to the next polygon for clustering. For this we need the next data point to be considered as the center point. This point can be taken from the farthest points of the merged triangles of the current polygon (bordering points). We find the farthest point of each merged triangle of the octagon. This can be done by finding the distance of all the points of the triangle from the octagon center using distance formulae. Hence, the points for which the distances are the maximum are the farthest points. Then we take one of them as our next center and continue. This goes on iteratively till we create octagons with all unclassified farthest points of merged triangles. After this, in 7 we repeat 2 through 6 till we can classify no more and we henceforth assign *cluster_id* to the cluster obtained. Then we repeat step 1 through 7 till the whole data set is classified and all clusters are formed. Fig. 10 visualizes the above mentioned stage.

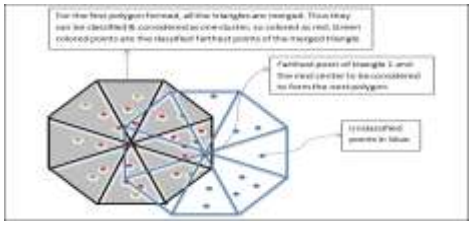

Figure 10- Formation of the next octagon (in blue) from one of the farthest points (in green)

***Procedure of TDCT:***

**The execution of the proposed algorithm includes the following steps:**

1. Creating the octagon taking an arbitrary unclassified point as center and epsilon distance ($\varepsilon$) as radius.
2. Compute the number of points in each triangle of an octagon (density).
3. Identify the maximum dense triangle of the octagon.
4. Traverse the neighboring triangles starting from the dense triangle in both directions and finding the ratio between two triangles w.r.t points in them.
5. Merging two triangles if ratio is greater than a certain threshold ($\beta$) and mark them as classified.
6. Find the farthest points of the merged triangles and creating the next octagon out of them.
7. Repeat step 2 through 7 till we can classify no more into a cluster and assign *cluster_id*.
8. Repeat step 1 through 8 till whole dataset is classified and all clusters are formed.

a) b)

Figure 11(a) & 11(b) - Application of polygon approach in dataset

## V. Results and performance analysis

To evaluate the technique in terms of quality of clustering, the algorithm was also applied on the Chameleon t4.8k.dat, t5.8k.dat, t8.8k.dat and t7.10k.dat datasets [9]. Fig. 12(a),12(b),12(c) & 12(d) shows the result of clustering in Chameleon t4.8k.dat, t7.10k.dat, t8.8k.dat and t5.8k.dat datasets respectively. The results obtained are shown in Fig. 15(*a*) and 15(*b*) respectively. From our rigorous experiments it has been found that the clustering result is dependent on the threshold $\beta$ which varies in the interval [0.4, 0.7]. The epsilon distance ε is set between 2.8 to 4 for optimized results.

a) b)

c) d)

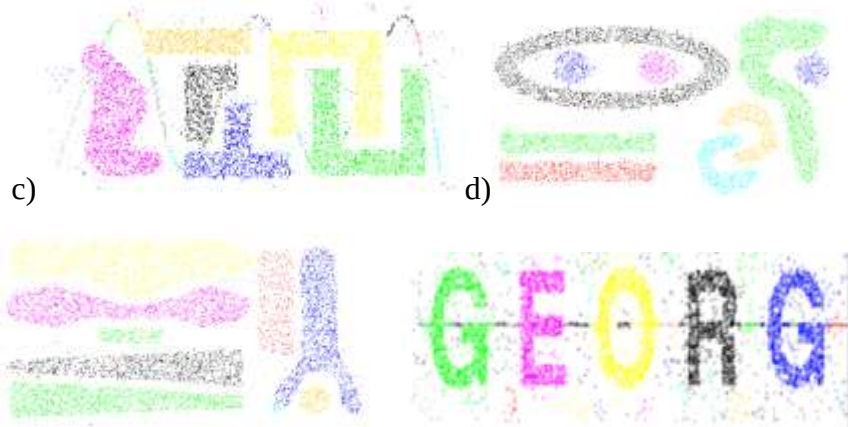

Figure 12(a),12(b),12(c) & 12(d)- Chameleon t4.8k.dat, t7.10k.dat, t8.8k.dat and t5.8k.dat datasets respectively

From the experimental results given above, we can conclude that TDCT is highly capable of detecting intrinsic as well as multi-density clusters qualitatively. However, using the idea of grid-based clustering along with TDCT can result in more accurate results which is again a future scope of research.

## VI. COMPLEXITY ANALYSIS

The partitioning of the neighborhood of an arbitrary data point into a polygon containing *m* non-overlapping triangles result in a complexity of $O(m)$. Finding of the densest triangle among *m* triangles within a polygon requires $O(m)$. The expansion of the cluster results in $O(N*m)$ time complexity, where *N* is the number of data points in the cluster formed and $m << N$ in the average case. If the number of clusters obtained is $n_c$ then, the overall time complexity for the clustering will be $O(n_c \times 2*m*N)$.

| Algorithms | No. of Parameters | Optimized for | Structure | Multi-Density Cluster | Embedded Clusters | Complexity | Noise Handling |
|---|---|---|---|---|---|---|---|
| K-means | No. of clusters | Separated Clusters | Spherical | No | No | $O(l^tkN)$ | No |
| K-modes | No. of clusters | Separated Clusters, Large Datasets | Spherical | No | No | $O(l^tkN)$ | No |
| PAM | No. of clusters | Separated Clusters, Large Datasets | Spherical | No | No | $O(l^tk(N-k)^2)$ | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| CLARA | No. of clusters | Relatively Large Datasets | Spherical | No | No | $O(ksz^2+k(N-k))$ | No |
| CLARANS | No. of clusters, Max no. of neighbors | Better than PAM, CLARA | Spherical | No | No | $O(kN^2)$ | No |
| BIRCH | Branching Factor, Diameter, Threshold | Large data | Spherical | No | No | $O(N)$ | Yes |
| ROCK | No. of clusters | Small noisy data | Arbitrary | No | No | $O(N^2+Nm_m m_a+N^2 logN)$ | Yes |
| CHAMELEON | 3($k$-nearest neighbors, MIN-SIZE, $\alpha^c$) | Small datasets | Arbitrary | Yes | No | $O(N^2)$ | Yes |
| DBSCAN | 2(*MinPts*, $\varepsilon$) | Large datasets | Arbitrary | No | No | $O(N \log N)$ using $R^*$ tree | Yes |
| OPTICS | 3(*MinPts*, $\varepsilon$, $\varepsilon'$) | Large datasets | Arbitrary | Yes | Yes | $O(N \log N)$ using $R^*$ tree | Yes |
| DENCLUE | 2(*MinPts*, $\varepsilon$) | Large datasets | Arbitrary | No | No | $O(N \log N)$ using $R^*$ tree | Yes |
| Wave Cluster | No. of cells for each dimension, No. of applications of transform | Any Shape, Large Data | Any | Yes | No | $O(N)$ | Yes |
| STING | No. of cells in lowest level, No. of objects in cell | Large spatial datasets | Vertical and horizontal boundary | No | No | $O(N)$ | Yes |
| CLIQUE | Size of the grid, minimum no. of points in each grid cell | High dimensional, Large datasets | Arbitrary | No | No | $O(N)$ | Yes |
| MAFIA | Size of the grid, minimum no. of points in each grid cell | High dimensional, Large datasets | Arbitrary | No | No | $O(c^{kl})$ | Yes |
| GDCT | 2 ($n$, $\beta$) | Large datasets | Arbitrary | Yes | Yes | $O(N)$ | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TDCT | 2($\varepsilon$, $\beta$) | Large datasets | Arbitrary | Yes | Yes | $O(n_c \square \times 2*m*N)$ | Yes |

Table 1: Comparison of the Proposed Algorithm (TDCT) with its counterparts

DBSCAN requires two input parameters MinPts and $\beta$. Moreover, it cannot detect embedded clusters. OPTICS on the other hand, requires three input parameters *MinPts*, $\varepsilon$ and $\varepsilon'$. But, it can detect embedded clusters. However, its performance degrades while detecting multiple nested clusters over massive datasets. Again, GDLC and Density-isoline algorithms can detect multi-density clusters but fail to detect intrinsic cluster structures. GDCT [14] requires the number of grid cells, i.e. *n* and threshold $\beta$ as input parameters. TDCT needs epsilon distance, $\varepsilon$ and threshold $\beta$ as input parameter Moreover, from our experiments we conclude that the threshold $\beta$ does not vary significantly with different datasets so it can be set beforehand and need not be entered by the user. The algorithm can effectively detect embedded clusters over variable density space as well as multiple nested clusters. A detailed comparison is given in Table 1.

***Advantages of proposed algorithm (TDCT)*:**

**The advantages of the proposed algorithm are:**

1) Embedded cluster Detection

2**)** $O(n_c \square \times 2*m*N)$ complexity

3) Handling of huge datasets

4) Handling of single linkage problem

5) Lesser number of parameters than its counterparts

## VII. CONCLUSION

This paper presents a clustering technique for massive numeric datasets. The clustering algorithm is based on density approach and can detect global as well as embedded clusters. Experimental results are reported to establish the superiority of the algorithm in light of several synthetic data sets. In this project we have only considered two-dimensional objects. But, spatial databases also contain extended objects such as polygons. Therefore, there is scope for scaling the proposed algorithm to detect clusters in such datasets with minor modifications, research of which is in progress. From a proper analysis of the designed technique, it can be safely concluded that the algorithm developed is working properly to a great extent.

## VIII. ACKNOWLEDGEMENTS

We take this opportunity to express our deep sense of gratitude and appreciation to The Department of Computer Science and Engineering, Jorhat Engineering College and The Department of Information Technology, Sikkim Manipal Institute of Technology for giving the required support and time without which we could not have made through it.

## REFERENCES

[1] J. Han and M. Kamber, *Data Mining: Concepts and Techniques.* India: Morgan Kaufmann Publishers, 2004.

[2] M. Ester, H. P. Kriegel, J. Sander and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", in *International Conference on Knowledge Discovery in Databases and Data Mining* (KDD-96), Portland, Oregon, 1996, pp. 226-231.

[3] C. Hsu and M. Chen, "Subspace Clustering of High Dimensional Spatial Data with Noises", *PAKDD*, 2004, pp. 31-40.

[4] W. Wang, J. Yang, and R. R. Muntz, "STING: A Statistical Information Grid Approach to Spatial data Mining", in *Proc.* 23rd *International Conference on Very Large Databases*, (VLDB), Athens, Greece, Morgan Kaufmann Publishers, 1997, pp. 186 - 195.

[5] G. Sheikholeslami, S. Chatterjee and A. Zhang, "Wavecluster: A Multiresolution Clustering approach for very large spatial database", in *SIGMOD*'98, Seattle, 1998.

[6] R. Agrawal, J. Gehrke, D. Gunopulos and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications", in *SIGMOD Record ACM Special Interest Group on Management of Data*, 1998, pp. 94–105.

[7] H. S. Nagesh, S. Goil and A. N. Choudhary, "A scalable parallel subspace clustering algorithm for massive data sets", in *Proc. International Conference on Parallel Processing*, 2000, pp. 477.

[8] L. Ertoz, M. Steinbach and V. Kumar, "Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data", in *SIAM International Conference on Data Mining* (SDM '03), 2003.

[9] G. Karypis, Han and V. Kumar, "CHAMELEON: A hierarchical clustering algorithm using dynamic modeling", *IEEE Computer*, 32(8), pp 68-75, 1999.

[10] Y. Zhao, S. Mei, X. Fan, S. Jun-de. 2003. Clustering Datasets Containing Clusters of Various Densities. *Journal of Beijing University of Posts and Telecommunications*, 26(2):42-47.

[11] H. S. Kim, S. Gao, Y. Xia, G. B. Kim and H. Y. Bae, "DGCL: An Efficient Density and Grid Based Clustering Algorithm for Large Spatial Database", *Advances in Web-Age Information Management* (WAIM'06), pp. 362-371, 2006.

[12] M. Ankerst, M. M. Breuing, H. P. Kriegel and J. Sander, "OPTICS: Ordering Points To Identify the Clustering Structure", in *ACMSIGMOD*, pp. 49-60, 1999.

[13] S. Roy and D. K. Bhattacharyya, "An Approach to Find Embedded Clusters Using Density Based Techniques", in *Proc. ICDCIT*, LNCS 3816, pp. 523-535, 2005.

[14] S. Sarmah, R. Das and D. K. Bhattacharyya, "Intrinsic Cluster Detection Using Adaptive Grids", in *Proc. ADCOM'07*, Guwahati, 2007.

[15] S. Sarmah, R. Das and D.K. Bhattacharyya, "A Distributed Algorithm for Intrinsic Cluster Detection over Large Spatial Data" A grid-density based clustering Technique (GDCT), World Academy of Science, Engineering and Technology 45, pp. 856-866, 2008.

[16] Rajib Mall ,"*Software Engineering*".

[17] Available: http//steve.hollasch.net /cgindex/math /barycentric.html