Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.1004 – 1012

SURVEY ARTICLE

Density-Based Spatial Clustering – A Survey

Naveen Kumar[1], S.Sivasathya[2]

[1,2]Department of Computer Science, Pondicherry University, India

[1] nav.bharti@gmail.com; [2] ssivasathya@gmail.com

Abstract— ***Spatial data mining is the task of discovering knowledge from spatial data. Density-Based Spatial Clustering occupies an important position in spatial data mining task. This paper presents a detailed survey of density-based spatial clustering of data. The various algorithms are described based on DBSCAN comparing them on the basis of various attributes and different pitfalls. The advantages and disadvantages of each algorithm is discussed.***

Keywords— ***Spatial Data Mining; Spatial Database; Spatial Clustering; Density-Based Data Mining; DBSCAN***

I. INTRODUCTION

Spatial data mining is the method of discovering interesting and previously unknown patterns from large spatial datasets, which includes *spatial classification*, *spatial clustering*, *spatial association rules* and *spatial outlier detection* etc. [11] Spatial clustering is the task of grouping a set of spatial objects or points into clusters so that objects within a cluster have high similarity among the spatial objects in comparison to one another, but are dissimilar to objects in other clusters. Basically spatial clustering is categorized into four different categories *partitioning method*, *hierarchical method*, *density-based method*, *grid-based method*.

Density-based spatial clustering [4] is based on the idea, that a set of spatial objects in higher density region should be grouped together into one cluster and a set of spatial objects in lower density regions are separated from the higher density region. These algorithms search for regions of higher density in a feature space that are separated by regions of lower density. The density-based methods can be used to filter out *noise*, and discover clusters of arbitrary shapes.

Density-based clustering algorithms are efficient and better in performance compared to hierarchical methods and partitioning method. This does not require the number of clusters priori as other algorithms like k-means. Also it works in the presence of obstacles and noise.

Discovering the knowledge from spatial data collected from the satellite images, radars, X-rays crystallography, military battlefields analysis, Geographical Information Systems (GIS), Global Positioning System (GPS), earth science (grouping earthquake epicenters to identify dangerous zones), biology (groupings of DNA sequences), image processing etc, are the main applications of density based spatial clustering. The rest of the paper is organized as follows. Section 2 presents related work. Section 3 provides Density-Based Spatial Clustering categorization. Section 4 gives the discussion of density-based spatial clustering and then comparison among density-based spatial clustering; finally section 5 concludes the paper.

II. RELATED WORK

As huge volumes of spatial data are collected from different sources every-day, spatial database systems have become popular during the last few years. A spatial database system provides spatial data types (SDTs) in their

data model, like POINT, LINE, REGION and also provide fundamental relationships (l intersects r), properties (area(r) > 1000), operations (intersection (l, r)) and query language support for spatial data types in its implementation. Apart from the database support, spatial indexing and efficient algorithms for spatial join are also given.

A. Spatial Clustering Method

There are four different methodologies for spatial clustering. They are: *Partitioning methods*, *Hierarchical methods*, *Density-based methods* and *Grid-based methods that are* implemented in SDBMS.

1) Partitioning methods [11] had long been popular clustering methods before the emergence of data mining. Given a set D of n objects in a d-dimensional space and an input parameter *k*, a partitioning algorithm organizes the objects into *k-clusters* such that the total deviation of each object from its cluster center or from a cluster distribution is minimized. The deviation of a point can be computed differently in different algorithms and is more commonly called a similarity function. Three different partitioning algorithms are available in literature.

- K-means algorithm
- EM (expectation maximization) algorithm
- K-medoids algorithm

The *k-means* [14] algorithm uses the mean value of the spatial objects in a cluster as the cluster center. The objective criterion used in the algorithm is typically the squared-error function. The EM (Expectation Maximization) [16] algorithm represents each cluster using a probability distribution. Typically, the Gaussian probability distribution is used because according to density estimation theory, any density distribution can be effectively approximated by a mixture of Gaussian distribution [17] functions. The k-medoids [15] method uses the most centrally located objects in a cluster to be the cluster center instead of taking the mean value of the objects in a cluster. Because of this, the k-medoids method is less sensitive to noise and outliers.

2) *Hierarchical methods* [12] create a hierarchical decomposition of the given set of spatial data objects forming a *dendrogram* – a tree which splits the database recursively into smaller subsets. The dendrogram can be formed in two ways: "*bottom-up*" or "*top-down*".

The "*bottom-up*" approach, also called the "*agglomerative*" approach, starts with each object forming a separate group. It successively merges the objects or groups according to some measures like the distance between the two centers of two groups and this is done until all of the groups are merged into one, or until a termination condition holds.

The "*top-down*" approach, also called the "*divisive*" approach, starts with all the objects in the same cluster. In each successive iterations, a cluster is split into smaller clusters according to some measures until eventually each object is in one cluster, or until a termination condition holds. AGNES and DIANA [18] are two earlier hierarchical clustering algorithms.

AGNES (AGglomerative NESting) is a bottom-up algorithm which starts by placing each object in its own cluster and then merging these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster or until a certain termination condition is satisfied. DIANA (DIvisive ANAlysis), on the other hand, adopts a top-down approach that does the reverse of AGNES by starting with all objects in one cluster. BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) [12] is an integrated hierarchical clustering method. The main concept of BIRCH is to compress the data objects into many small subclusters and then perform clustering with these subclusters. Due to the compression, the number of subclusters is much less than the number of data objects and thus it allows the clustering to be performed in the main memory. This gives result in an algorithm which only needs to scan the database once.

The CURE [19] is an agglomerative method which uses a more sophisticated principle when merging clusters. Two main ideas imply to obtain high quality clusters. First, instead of using a single centroid or object to represent a cluster, a fixed number of well-scattered objects are selected to represent each cluster. Second, the selected representative objects are shrunk towards their cluster centers by a specified fraction called shrinking factor α which ranges between [0, 1].

Similar to CURE, CHAMELEON [20] is a clustering algorithm which tries to improve the clustering quality by using more elaborate criteria when merging two clusters. Two clusters will be merged if the inter-connectivity and closeness of the two individual clusters are very similar.

3) *Density-based method:* It typically regards clusters as dense regions of objects in the data space which are separated by regions of low density (representing noise). Density-based methods can be used to filter out noise (outliers), and discover clusters of arbitrary shape. [8] DBSCAN is one such algorithm which grows regions with sufficiently high density into clusters, and discovers clusters of arbitrary shape in spatial databases. The algorithm requires the input of two parameters ε and *MinPts;* where ε is the radius of the cluster and MinPts is the minimum number of points allowed in the cluster. The neighborhood within a radius ε of a given object is called the ε-neighborhood having a core object.

OPTICS (Ordering Points To Identify the Clustering Structure) [21] is an improvement to the DBSCAN wherein it orders the input points for clustering, this also needs input ε and *MinPts*. The OPTICS algorithm creates an ordering of the objects in a database, additionally storing the core-distance and a suitable reachability-distance for each object. Such information is sufficient for the extraction of all density-based clustering with respect to any distance ε' that is smaller than the distance *ε* used in generating the order. *DENCLUE* (DENsity-based CLUstEring) [22] is based on a set of density distribution functions, influence function, which describes the impact of a data point within its neighborhood. The overall density of data space is modeled analytically as the sum of the influence function of all data points. The cluster is determined by density attractors, where density attractors are local maxima of the overall density function.

4) *Grid-based method:* It uses a gird data structure; it uses the space for finite number of cells which form a grid structure on which all of the operations for clustering are performed. The algorithm works fast and is independent of the number of data objects. *STING* (STatistical INformation Grid) [23] is a multi-resolution data structure in which spatial area is divided into rectangular cells. There are usually several levels of such rectangular cells corresponding to different levels of resolution, and these cells form a hierarchical structure. Each cell at a high level is partitioned to form a number of cells at the next lower level. *WaveCluster* [13] is a multi-resolution clustering algorithm which first summarizes the data by imposing a multidimensional grid structure onto the data space. It then uses the *wavelet transformation* to transform the original feature space, finding dense regions in the transformed space. Wavelet transform is a signal processing technique that decomposes a signal into different frequency sub-bands that can be applied to n-dimensional signals by applying a one-dimensional wavelet transforms n number of times. The *CLIQUE* algorithm is a combination of density-based and grid-based clustering. The data space is partitioned into non-overlapping rectangular units by equal space partition along each dimension. A unit is dense if the fraction of total data points contained in it exceeds an input model parameter; a cluster is defined as a maximal set of connected dense units.

III. DENSITY-BASED SPATIAL CLUSTERING

The focus of this survey is on this section which presents the different types of algorithms that are categorized under Density-based spatial clustering.

A. DBSCAN (Density-Based Spatial Clustering of Application with Noise)

DBSCAN [4] is a density-based clustering algorithm which is designed to discover clusters and noise of spatial objects in spatial database. It is necessary to know the parameters ε and *MinPts* of different clusters and at least one point from each cluster. The ε (epsilon) is radius of the cluster and MinPts is the minimum number of points in the cluster. Algorithm finds point p and density-reachable points from p with respect to ε and MinPts. The DBSCAN algorithm relies on density-based notions of cluster. These are defined as:

- **ε-neighborhood of point ($N_\varepsilon(p)$):** The ε-neighborhood of a point p, is the set of point objects in the diameter of ε.

$N_\varepsilon(p) = \{q \in D | \text{dist}(p, q) \leq \varepsilon\}$.

Where ε is the diameter of the cluster and dist(p, q) is the distance function for two points p and q.

- **Directly density-reachable:** For every point p in a cluster C there is a point q in C so that p is inside of the ε-neighborhood of q and $N_\varepsilon(q)$ contains at least MinPts points.

i) $p \in N_\varepsilon(q)$ and

ii) $|N_\varepsilon(q)| \geq$ MinPts (core point condition).

- **Density-reachable:** A point p is density-reachable from a point q with respect to ε and 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.
- **Density-connected:** A point p is density-connected to a point q with respect to ε and MinPts if there is a point o such that both, p and q are density-reachable from o with respect to ε and MinPts.
- **Cluster:** Let D be a database of points. A cluster C with respect to ε and MinPts is a non-empty subset of D satisfying the following conditions:

i) ∀ p, q: if $p \in C$ and q is density-reachable from p with respect to ε and MinPts, then $q \in C$. (Maximality)

ii) ∀ p, q ∈ C: p is density-connected to q with respect to ε and MinPts. (Connectivity)

- **Noise:** Let $C_1,\ldots,C_k$ be the clusters of the database D with respect to parameters εi and $MinPts_i$, i=1,…,k. Then the noise is defined as the set of points in the database D not belonging to any cluster C_i, i.e. noise = $\{p \in D \mid \forall i: p \notin C_i\}$.

B. An Adaptive DBSCAN

DBSCAN algorithm should contain at least one core object to define a cluster in spatial database. *MinPts* must be large enough to distinguish clusters and noise. In [3] the author has defined two measures for quality of density in DBSCAN: 1) *Density pad*: A density pad is a convex region inside a circle with radius *Eps* that includes all the useful objects. 2) *Void pad*: A void pad is the region inside the circle with radius *Eps* that is not density pad. Larger void pad become the reason for noise data in a cluster, which causes chain affection, that result in lower quality of density. Void pad can be minimized in two ways to achieve better quality of density. 1) By setting *Eps* as small as possible, and large enough to avoid clusters to be divided into sub-clusters or noise. 2) Using different measures to select an object's neighbors that help in improving the quality of density. The method selection to define neighboring region improve the clustering accuracy by avoiding void pad in density-based clustering algorithms. In the Adaptive DBSCAN [3] two types of relationship are defined.

1) ***Directly border-reachable*:** An object p is directly border-reachable from an object q with respect to *MinPts* and ε if

- p is a border object ($|N_\varepsilon(p)|<MinPts$),
- q is a core object ($|N_\varepsilon(q)| \geq MinPts$),
- $p \in N_\varepsilon(q)$, where $N_\varepsilon(q)$ denotes the neighboring region, $|N_\varepsilon(q)|$ denotes the number of objects in the region.

2) ***Relationship between two core objects:*** (directly core connected or core connected)

***Directly core-connected*:** A core object p is directly core-connected to a core object q with respect to *MinPts* and ε, if $p \in N\varepsilon(q)$ and $q \in N_\varepsilon(p)$.

***Core-connected*:** An object p is core-connected to an object q with respect to MinPts and ε, if there is a chain of objects $p_1,\ldots,p_n$, $p_n=q$ such that p_{i+1} is directly core-connected from p_i.

Based on these three definition density-based notion of cluster is defined as core-connected to each other in a cluster and directly reachable by at least one core object in the cluster.

Based on the above two relationships, a cluster in Adaptive DBSCAN is defined as follows

***Cluster*:** Let D be a set of objects. A cluster C with respect to MinPts and ε is a non-empty subset of D satisfying the following conditions:

- Maximality: ∀p, if there is a core object $q \in C$ and p is directly border-reachable from q or p is core-connected with q with respect to ε and MinPts, then $p \in C$.
- Connectivity: ∀p,q ∈ C, if p and q are core objects, then p core-connected to q with respect to ε and MinPts.
- Reachability: ∀p in C, if p is border object, there must have a core object q in C and p is directly border-reachable from q with respect to ε and MinPts.

C. LDBSCAN (Local Density-Based Spatial Clustering of Application with Noise)

In many cases DBSCAN algorithm is not suitable because of its global density parameters in class identification of spatial database, where local-density clusters exist. The parameters used by clustering algorithms are hard to determine but have significant influence on the clustering results. In [4] LDBSCAN algorithm relies on the local density-based notion of clusters and overcomes the above problems taking advantage of LOF (local outlier factor). The LOF represents the degree of each object that outlies and LRD (local reachability density) represents the local density of the object.

It is very easy to pick the appropriate parameters LOFUB, pct and MinPts of clusters and one core point of the respective cluster. The parameter LOFUB (local outlier factor upper bound) is upper-bound of LOF and the parameter *pct* is used to control the fluctuation of local-density, local-density-reachable. Then local-density-reachable points from the core point are retrieved using correct parameters. If arbitrary selected point p is a core point a cluster is formed. If p is not a core point LDBSCAN checks for the next point of the database. LDBSCAN is based on the following notions of clusters:

- **Core point:** A point p is a core point with respect to LOFUB if $LOF(p) \leq LOFUB$.
- **Directly local-density-reachable:** A point p is directly local-density-reachable from a point q with respect to pct and MinPts if

i) $p \in NMinPts(q)$ and

ii) $LRD(q)/(1+pct) < LRD(q)*(1+pct)$

- **Local-density-reachable:** A point p is local-density-reachable from the point q with respect to pct and MinPts if there is a chain of points $p_1,p_2,\ldots,p_n$, where $p_1=q$, $p_n=p$ such that p_{i+1} is directly-density-reachable from p_i.

- **Local-density-connected:** A point p is local-density-connected to a point q from o with respect to pct and MinPts if there is a point o such that both p and q are local-density-reachable from o with respect to pct and MinPts.
- **Cluster:** Let D be a database of points, and point o is a selected core point of C, i.e. $o \in C$ and LOF (o) $\leq$ LOFUB. A cluster C with respect to LOFUB, pct and MinPts is a non-empty subset of D satisfying the following conditions:

i) $\forall p$: p is local-density-reachable from o with respect to pct and MinPts, then $p \in C$. (maximality).
ii) $\forall p, q \in C$: p is local-density-connected q by o with respect to LOFUB, pct and MinPts. (connectivity).

- **Noise:** Let $C_1,\ldots,C_k$ be the clusters of the database D with respect to parameters LOFUB, pct and MinPts. Then noise = $\{p \in D \mid \forall i : p \in C_i\}$.

D. GRIDBCAN (GRId Density-Based Clustering of Application with Noise)

One of the major problems of DBSCAN is its inability to recognize clusters with different densities in spatial database. This problem is addressed by [6] GRIDBSCAN through three levels of clustering methods. In the first level the algorithm selects appropriate grids for homogeneous density in each grid, in the second level it merges similar densities cells and indentifies input parameters ε and MinPts for each grid, then in the third level DBSCAN main method is executed with the identified parameters ε and MinPts in the database.

GRIDBSCAN defines an upper and lower bound for the data in cells to decide that the cell is dense enough for further process. Any cell that has less than lower bound data points is not to be processed and the cell that contains data in the interval (lower, upper) bound is treated as sparse cells that are worthwhile for further processes. Otherwise cells will be considered to be data dense. This procedure is repeated until either all cells are processed or cells contain no data vector. The algorithm then connects cells with similar ε values to distinguish those regions with different densities. After merging the cells of similar ε values the attributes of the merged cells are updated. Finally DBSCAN algorithm is run on the data with different density parameters. The main module is run for every unprocessed data vector in each cell by using the corresponding density parameters.

The complexity of GRIDBSCAN in the second stage is the most time consuming part, due to the fact the algorithm uses the DBSCAN while merging the cells. The GRIDBSCAN algorithm is useful for small and medium size spatial databases but not for large spatial databases.

E. VDBCAN (Varied Density-Based Clustering of Application with Noise)

Many existing density-based algorithms have drawbacks in finding clusters for datasets with varied densities. VDBSCAN [1] is proposed for the purpose of varied-density datasets analysis. Before applying DBSCAN algorithm several values of ε is selected for different densities according to k-dist plot, it is possible to find out clusters with varied densities using different values of ε.

The algorithm monitors the behaviour of the distance from a point to its Kth nearest neighbour (k-dist) to determine parameters ε and MinPts. The k-dist is the value computed for all the data points for some k, plotted in ascending order.

The new algorithm VDBSCAN which is an improved version of DBSCAN works as follows

1) First it calculates and stores k-dist for each object and partition k-dist plots.
2) Second, the number of densities is given by k-dist plot.
3) Third, choose parameters ε_i automatically for each density.
4) Fourth, scan the datasets and cluster different densities using corresponding ε_i. And finally, form the valid clusters corresponding to varied densities.

F. LD-BSCA (A Local-Density Based Spatial Clustering Algorithm)

The basic idea of DBSCAN is that for each point of a cluster the neighbourhood of a given radius ε has to contain at least a MinPts points otherwise DBSCAN can't form accurate cluster. Most of the density-based clustering algorithm is unable to characterize the intrinsic cluster structures by global density parameter. In case of varied local-density clusters that exist in different regions of data space a new algorithm LD-BSCA is proposed with the concept of local MinPts and new cluster expanding condition *ExpandConClId.*

LD-BSCA [5] algorithm relies on the notions fall, local MinPts and ExpandConDlId designed to discover the clusters in a spatial database. Let D include n data objects, the algorithm preprocess the database in increasing order of degrees in an appointed dimension. It starts with the smallest point p and retrieves all objects local-density-reachable from p with respect to eps and *ExpandConClId.* If p is unclassified then it is a core point, and in this case LD-BSCA creates a new cluster.

Some of the basic concepts of LD-BSCA are as follows:

- **Core objects:** A core object is an unclassified point.
- **eps-neighborhood Neps(p) of a point:** eps-neighborhood of a point is defined as $N_\varepsilon(p) = \{q \in D | \text{dist}(p, q) \leq \varepsilon\}$.

- **Directly local-density-reachable:** An object q is directly local-density-reachable from an object p if q $\in N_\varepsilon(p)$, and p is a core object.
- **ExpandConClId:** Let p be a core point of the ClId-th cluster C, ClId is a numeral value. The cluster C can be expanded if one of the objects q satisfies the following condition:
  $q \in \{q \in N_\varepsilon(p)$, q not equal p$\}$
  $CN_\varepsilon MinPtsClId(p) \leq CN_\varepsilon MinPtsClId(p)(1+(CN_\varepsilon MinPtsClId(p))-1/2)$
- **Local-density-reachable:** An object p is local-density-reachable from the object q with respect to ε and ExpandCondDlId if there is a chain of objects $p_1,\ldots p_n$, p_1= p and p_n=q such that p_{i+1} is directly density-reachable from p_i with respect to and ExpandConDlId, for $1 \leq i \leq n$, $p \in D$.

G. EDBSCAN (An Enhanced Density-Based Spatial Clustering of Application with Noise)

One of the main problems of DBSCAN is having wide density variation within a cluster. The algorithm does not apply upper limit for core object. The Enhanced DBSCAN [7] overcomes the problem mentioned, based on the concept that it calculates the density variation of a core object with respect to the densities of all its ε-neighborhood.

EDBSCAN starts clustering by selecting the Core object; it inserts the selected Core object into the Queue. It pops out the object from the front from the seed list (Queue). It calculates all the ε-neighborhood of the Core object and finds out all the Relative Density Core objects. It inserts all the Relative Density Core objects for further expansion in the Queue, if still unclassified. The rest of the objects which are not Relative Density Core with their surrounding, are simply added into the cluster, still unclassified. It expands all the Relative Density Core objects one by one, popping out from the Queue, by following the above described procedures, which contribute more Relative Density Core objects for further expansion. This is repeated until the queue is empty and the entire cluster is computed. So finally all the objects are either assigned a certain clusterID or marked as Noise.

H. Improved VDBSCAN (Improved Varied Density-Based Spatial Clustering Algorithm with Noise)

The main problem with VDBSCAN is that it depends on input parameters ε and MinPts. To solve the dependency of VDBSCAN on the input parameters a new and improved VDBSCAN [2] is introduced that selects parameters automatically for perfect clustering. The Improved VDBSCAN starts with partitioning the dataset using Euclidean distance measure. Next the calculation of difference between the minimum points and the maximum points is done. Then the boundary is calculated with radius ε for every dataset. In the next step cluster will be classified for each such boundary. Finally all the clusters are formed and displayed.

Steps involved in the proposed algorithm:

- Partition the dataset using Euclidean-distance.
- For Minpts, calculate the difference between the minimum point and the maximum points.
- Calculate ε for each partition dataset.
- For each ε adopt DBSCAN algorithm.
- Mark points as C_{i-t} (C_{i-t} is ith cluster).
- Display the marked points as corresponding clusters.

I. Density-Based Clustering of Polygons

One of the clustering algorithms based on DBSCAN is P-DBSCAN [9] which is to cluster polygons. The key factor in this algorithm is to incorporate the topological and spatial properties in the process by using distance function customized for the polygon space. This algorithm directly does not apply to polygon, so density-based concepts for Polygons are defined like *core polygons* and its *neighbourhood*. The rest of the concepts are as in DBSCAN and can be applied to polygons.

Density-based concepts of polygon:

- **ε-neighborhood of a Polygon:** designed by $N_\varepsilon(p) = \{q \in D \mid dist(p, q) \leq \varepsilon\}$, where D is the dataset of Polygon, and dist(p, q) is a distance function between polygon p and polygon q.
- **Radial Spatial Neighborhood of a Polygon:** $N_\varepsilon(p) = U^R_{i=1} N_{\varepsilon,i}(p)$, such that R is number of equal-size sectors radially partitioning the space around the polygon p.
- **Core Polygon:** A core polygon c is a polygon that has a minimum number of polygon (MinPolys) within its ε-neighborhood, and at least minimum number of radial spatial partitions (MinS).
- **Border Polygon:** A border polygon b is a polygon that has more than R-MinS of its radial spatial partitions empty.
- **Outlier Polygon:** The polygon that does not have polygon within its threshold distance ε.

- **Direct Density-Reachable:** A polygon p is directly density-reachable from a polygon q with respect to ε, if
  a. $p \in N\varepsilon(p)$ and
  b. q is a core polygon.
- **Density-Reachable:** A polygon p is density-reachable from a polygon q if there is a chain of polygons $p_1,\ldots,p_n$ where $p_1=q$ and $p_n=p$ such that p_{i+1} is directly density-reachable from p_i, and i=1 to n.
- **Density-Connected:** A polygon p is density-connected to a polygon q if there is a polygon o such that both, p and q are density-reachable from o.
- **Cluster:** A cluster C with respect to ε is non-empty subset of D satisfying the following:
  a. **Maximality:** ∀ p, q: if $p \in C$, $q \in D$ is density-reachable from p then $q \in C$.
  b. **Connectivity:** ∀ p, q: if $p \in C$: p is density-connected to q.

Hausdroff distance function between two sets of points is defined as the maximum distance of points in one set to the nearest point in the other set. $D_h(A, B) = \max_{a\in A}(\min_{b\in B} d(a, b)), d(a, b)$. The distance metric between two points a and b is Euclidian distance. This algorithm works similar to DBSCAN, that it selects polygon $p \in D$ and $p \notin C$. In the case of unclassified polygon ExpandCluster method is called, in this method cluster assignment is done. If polygon is core polygon, its neighbors are assigned to the same cluster. The time complexity of P-DBSCAN is same as DBSCAN, i.e. O(nlogn), and n is the size of the database, P-DBSCAN results more compact clusters in comparison to DBSCAN.

J. GDBSCAN (Generalized Density-Based Clustering of Application with Noise)

This is the generalization [10] of DBSCAN which can cluster points and spatial objects on the basis of spatial and non-spatial attributes. The algorithm uses NPred-neighborhood, MinCard and wCard.

Let NPred be a binary predicate on database D which is reflexive and symmetric. Then the NPred-neighborhood of an object $o \in D$ is defined as $N_{NPred}(o) = \{o' \in D \mid NPred(o, o')\}$ such that, for all $p, q \in D$: NPred(p, p) and, if NPred(p, q) then NPred(q, p) [reflexive and symmetric].

MinWeight of a set of objects: The predicate MinWeight for a set S of objects is defined to be true if and only if $wCard(S) \geq MinCard$. Where *wCard* is a function from the powerset of database D into the non-negative Real Numbers, *wCard*: $2^D \rightarrow R^{\geq 0}$ and *MinCard* is a positive real number.

GDBSCAN chooses an arbitrary object p and finds all density-reachable objects from object p with respect to *NPred* and *MinWeight*, in the case of p being a core object. The algorithm has density-connected set with respect to *NPred* and *MinWeight*. In the case of p not a core object, p is assigned to **Noise**. This procedure is repeated until each object p has not yet been classified, thus clustering and noise is detected in this way.

Table-I gives a comparison of all the discussed density based clustering algorithms.

TABLE I
COMPARISON OF DENSITY-BASED ALGORITHMS

S. No.	Algorithm Name	Complexity	Input Parameter	Dataset Used
1	DBSCAN	O(nlogn)	Gloabal eps, MinPts	Synthentic & real dataset
2	ADBSCAN	O(n*runtime of neighborhood query), O(nlogn)- R*tree	No parameter	Different datasets, GPS, Military randomly
3	LDBSCAN	O(n*runtime neighborgood query), O(n)-2^{nd} step	No Parameter	Generated random datasets
4	GRIDBSCAN	O(NCM*NDlogND), NCM-no. of candidates; ND-no, of data vector	N/A	Three artificial benchmark dataset
5	VDBSCAN	N/A	Depend on parameter	Several databases
6	LD-BSCA	Much less than DBSCAN O(nlogn)	N/A	Different databases, Hang Zhou's park data
7	EDBSCAN	N/A	N/A	2-D Synthetic datasets
8	Improved VDBSCAN	Time is directly proportional to size of dataset	Automatic parameter selection	2-D synthetic datasets

9	P-DBSCAN	O(nlogn)	Eps, MinPolys	Nebraska census tract dataset, South Dakota census tract dataset
10	GDBSCAN	O(n2)-without index, O(nlogn)-with spatial index, O(n)-with direct access	NPred, wCard, MinCard	US Geological Survey data, SEQUIOA 2000 point data

IV. **CONCLUSION**

Density-based spatial clustering algorithms are very important in spatial data mining. This survey paper, deals with the Density-Based Spatial Clustering algorithms based on the basic algorithm DBSCAN. Various significant concepts related to spatial data mining and spatial clustering are discussed. Also, the basic classification of spatial clustering is described with example algorithms. Partitioning methods organize the objects in k-clusters using similarity function. Hierarchical method decomposes set of objects forming tree data structure or dendrogram. Density-based method separates regions into dense and low dense clusters. Grid-based method uses grid data structure to perform clustering task.

The main focus is given to the density-based spatial clustering with their working characteristics. DBSCAN which is designed to discover clusters and noise depends on epsilon and MinPts. An Adaptive DBSCAN defines two significant measures density-pad and void-pad for quality of density in DBSCAN. LDBSCAN and LD-BSCA give very good performance where local-density and varied local-density clusters exit respectively. GRIDBSCAN, VDBSCAN and EDBSCAN are useful where clusters are with different density variations. Improved DBSCAN does not take input parameter but finds them based on Euclidean distance measure. P-DBSCAN is the most important algorithm that incorporates topological and spatial properties for polygon space clustering. GDBSCAN is the generalization of DBSCAN that clusters points on the basis of spatial and non-spatial attributes. Finally all the algorithms are compared and tabulated.

REFERENCES

[1] Peng Liu; Dong Zhou and Naijun Wu, *VDBSCAN: Varied Density-Based Spatial Clustering of Applications with Noise*, 1-4244-0885-7/07, 2007 IEEE.

[2] Vijayalakshmi and Dr. M. Punithavalli, *Improved Varied Density-Based Spatial Clustering Algorithm with Noise*, 978-1-4244-5967-4/10, 2010 IEEE.

[3] Daoying, Ma and Aidong Zhang, *An Adaptive Density-Based clustering Algorithm for Spatial Database with Noise*, Proceedings of the Fourth IEEE International Conference on Systems, Man and cybernetics.

[4] Lian Duan; Deyi Xiong; Jun Lee and Feng Guo, *A Local Density Based Spatial Clustering Algorithm with Noise*, 2006 IEEE International Conference on Systems, Man, and Cybernetics.

[5] Guiyi Wei and Haiping Liu, *LD-BSCA: A Local-Density Based Spatial Clustering Algorithm*, 978-1-4244-2765-9/09 2009 IEEE.

[6] Ozge Uncu; William A. Gruver; Dilip B. Kotak; Dorian Sabaz; Zafeer Alibhai and Colin Ng, *GRIDBSCAN: GRID Density-Based Spatial Clustering of Application with Noise*, 2006 IEEE Int. Conference on Systems, Man and Cybernetics, 1-4244-0100-3/06 2006 IEEE.

[7] Anant Ram; Ashish Sharma; Anand S. Jalal; Raghuraj Singh; and Ankur Agrawal, *An Enhanced Density Based Spatial Clustering of Application with Noise*, 2009 IEEE (IACC 2009), 978-1-4244-2928-8/09.

[8] Marin Ester; Hans-Peter Kriegel; Jorg Sander and Xiaowei Xu, *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*, Proc. of 2nd Int. Conference on (KDD-96).

[9] Deepti Joshi; Ashok K. Samal and Leen-Kiat Soh, *Density-Based Clustering of Polygons*, 978-1-4244-2765-9/09 2009 IEEE.

[10] Jorg Sander; Martin Ester; Hans-Peter Kriegel; Xiaowei Xu, *Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications*.

[11] Jiawei Han; Michelin Kamber and Anthony K. H. Tung, *Spatial Clustering Methods in Data Mining: A Survey*.

[12] Zhiwei SUN, *A Hierarchical Clustering Algorithm Based on Density for Data Stratification*, 2012 International Conference on Systems and Informatics (ICSAI 2012).

[13] Sheikholeslami, Gholamhosein, Surojit Chatterjee, and Aidong Zhang. "WaveCluster: a wavelet-based clustering approach for spatial data in very large databases." *The VLDB Journal* 8.3-4 (2000): 289-304.

[14] MacQueen, James. "Some methods for classification and analysis of multivariate observations." *Pro of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. No. 281-297. 1967.

[15] Kaunfman, L. and Rouseeuw, P.J. (1987), Clustering by means of Medoids, in Statistical Data Analysis Based on the L1-Norm and Related Methods, edited by Y. Dodge, North-Holland, 405-41.
[16] Dempter, A.P.; Laird, N.M.; Rubin, D.B., (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Statistical Society, Series BN 39(1): 1-39. JSTOR 2984875. MR0501537.
[17] Ribeiro, Maria Isabel. "Gaussian probability density functions: Properties and error characterization." *Instituto Superior Tcnico, Lisboa, Portuqal, Tech. Rep*(2004): 1049-001.
[18] Kaufman, Leonard, and Peter J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Vol. 344. John Wiley & Sons, 2009.
[19] Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim. "CURE: an efficient clustering algorithm for large databases." *ACM SIGMOD Record*. Vol. 27. No. 2. ACM, 1998.
[20] Karypis, George, Eui-Hong Han, and Vipin Kumar. "Chameleon: Hierarchical clustering using dynamic modeling." *Computer* 32.8 (1999): 68-75.
[21] Ankerst, Mihael, et al. "Optics: Ordering points to identify the clustering structure." *ACM SIGMOD Record*. Vol. 28. No. 2. ACM, 1999.
[22] Hinneburg, Alexander, and Hans-Henning Gabriel. "Denclue 2.0: Fast clustering based on kernel density estimation." *Advances in Intelligent Data Analysis VII*. Springer Berlin Heidelberg, 2007. 70-80.
[23] Wang, Wei, Jiong Yang, and Richard Muntz. "STING: A statistical information grid approach to spatial data mining." *VLDB*. Vol. 97. 1997.

