International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 1, January - 2014

# Spatial Databases and its Mining

Garima
*Computer Science & Engineering*
*Amity University*
*Noida, India*

Sangeeta Rani
*Assistant Professor*
*Amity University*
*Noida, India*

*Abstract*—**Spatial data mining is the process in which we extract interesting i.e. previously unknown and useful information from large spatial datasets. Extracting the useful and interesting patterns from the spatial datasets is more difficult due to the complexity of the spatial data types and spatial relationships. There has been a tremendous increase in the amount of spatial data and its application to different fields like in remote sensing, medical sciences, computer cartography, geographic information systems etc.**
**Many techniques have been designed for the spatial data mining like decision tree and classification. Clustering is the most widely used technique for the spatial data mining.**

***Index Terms*— with respect to, database management system, density based spatial clustering of applications with noise, varied density based spatial clustering of applications with noise, partitioning around medoids.**

## I. INTRODUCTION

Mining is defined as the process to extract useful and implicitly stated information. Mining can be performed on different types of databases like **relational databases** (which consist of a collection of interrelated data in the form of tables and a set of software programs to manage and access the data), **transactional databases** (which consists of the files where each record represents a transaction and a transaction includes a unique transaction id and a set of items), **temporal databases** (that store the relational data involving time related attributes like timestamps), **sequence databases**( that store the sequence of the ordered events with or without the time related attributes), **spatial databases** (that contain space related information like geographic databases, very large-scale integration (VLSI) or computer-aided design databases), **time series databases** (that store sequences of events obtained over repeated intervals of time like weekly, monthly, hourly etc).

Data mining includes the techniques and methods from various fields like machine learning, database systems, statistics etc. A major challenge in spatial data mining is the efficiency of the algorithms present for the spatial data mining due to the presence of large amount of data related to space. Spatial data mining methods are applied in order to extract useful and interesting information from large spatial databases. Spatial data mining can also be used to identify relationships between spatial and non spatial data, query optimization and data reorganization in spatial databases.

Our focus in this paper is on spatial databases and spatial data mining in order to understand the various spatial data mining methods available, their applicability in different situations and their strengths and weaknesses. The different algorithms proposed for spatial data mining have been presented.

## II. SPATIAL DATABASES

Spatial data is the data related to objects that occupy space. Space is like a framework that identifies the relationships among the set of objects. There are different models of space like Euclidean space, set based space, topological space, metric space etc. The space can be any 3 dimensional spaces representing the arrangement of chain of protein molecules, 2 dimension representation of surface of earth etc. Spatial data includes the information related to the topology, distance etc which is organized by the use of spatial indexes and is accessed through spatial access methods. Thus, making it more challenging for mining information from spatial data. A spatial database system is such a database where space related information is always connected with alphanumeric data, has spatial data types like POINT, LINE, and REGION etc in its data model and makes the use of spatial indexes for spatial join in query processing.

Spatial database is used to store and query data that represents objects defined in geometric space. The objects can be simple geometric objects like points, line and polygon or complex objects like 3D objects, topological coverage etc. Spatial databases use spatial index to speed up the query processing. One can perform the operations related to the computation of length of line, area of polygon, true or false queries related to the relationships among spatial objects etc. Various spatial index method used are Grid, Quadtree, Octree, R tree, m tree etc. In object relational DBMS the architecture of spatial databases consist of a spatial database component and the interface between the spatial database component and the application that maps the constructs to database.
Spatial databases focus on the space taxonomy, spatial data models and spatial query languages. The three level architecture of spatial database is shown in figure 1 below.

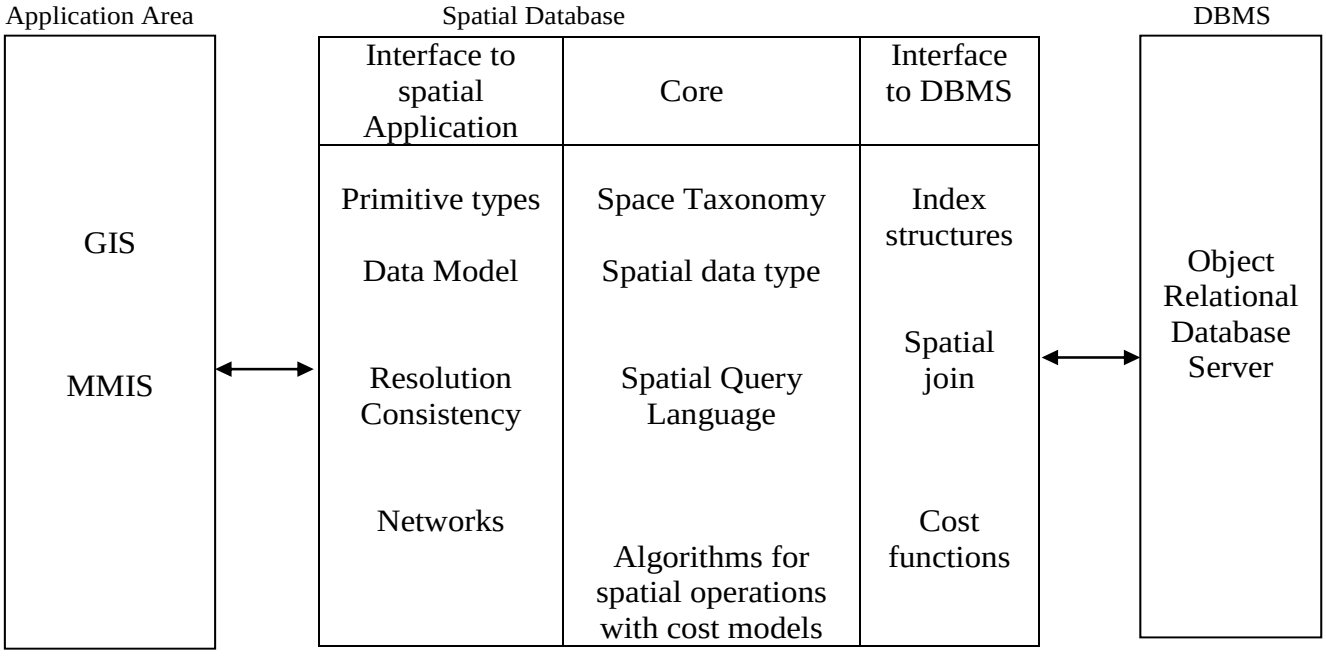


Fig.1 Three layer architecture of spatial database

## III. SPACE TAXONOMY

The most promising task in research towards spatial databases is to decide what needs to be represented i.e. either we need to represent the distinct entities having all the features or some features about each and every point in space.
Different models of space like Metric space, Euclidean space, set based space, topological space have been identified to establish relationships among spatial objects.

- Euclidean space converts the spatial properties and relationships to real numbers.
- Metric space uses the positive symmetric functions to identify the distance relationships.
- Set based space is used to identify the relationships like set equality, union, intersection and subset and is commonly used in relational and object – relational databases.

Topological space is used to identify the extended object relationships like boundary, interior, open, closed using the neighboring points.

## IV. SPATIAL RELATIONSHIPS AND DATA MODAL

There are several types of spatial relationships that have been categorized so that query processing becomes easier like:

a.) Topological relationships: these include adjacent, inside, disjoint etc.

b.) Direction relationships: directions like top, bottom are described as above, below etc.

c.)Metric relationships: like distance<100 etc.

Spatial data types can be integrated with the relational model or the ER model but it becomes difficult to handle partition and network. There are two commonly used data models in spatial databases – field based and object based. The field based spatial data model deals with the information like height or altitude of an object, rainfall and temperature etc. The Object based data model deals with the discrete entity based information and typical operations include distance and boundary.

## V. SPATIAL DATATYPES

Spatial data types are used to store the information about spatial objects. There are different types of algebras present for spatial objects like ROSE algebra. In ROSE algebra, three fundamental or predefined data types are present POINTS, LINES and REGIONS whose values are based on the objects from real world. We need to understand the R-block and R-Face in order to describe these values. R-block is a connected set of line segments of R and R- face is a polygon with holes thereby forming the value of *points* as a set of R points, *lines* as a disjoint set of R blocks and *regions* as a set of edge-disjoint R faces i.e. no common edge between the faces.

Suppose there are two sets EXT= {lines, regions} and GEO= {points, lines, regions} then there are three classes of operations like:

1.) Operations that express topological relationships:

$\forall$ *geo* in GEO. $\forall$ *ext*1, *ext*2 in EXT.

| | | | |
|---|---|---|---|
| *geo* × regions | → | bool | **inside** |
| ext1×ext2 | → | bool | **intersect** |

In the above example the variable geo can be of three types according to the kind GEO so that in **inside** operation we can compare any of the points, lines or regions value with a regions value. The **intersects** or meets operation is applicable to two lines variables or two regions variable.

2.) Operations that return fundamental data type values:

$\forall$ *geo* in GEO.

| | | | |
|---|---|---|---|
| *lines* × *lines* | → | points | **intersection** |
| *regions*× *regions* | → | regions | **intersection** |
| *geo* × *geo* | → | *geo* | **plus** |

Here **plus** represent the union and minus is also there to represent difference of two values of the same type.

3.) Operations that return numbers:

$\forall$ *geo*1 × *geo*2 in GEO.

| | | | |
|---|---|---|---|
| *geo*1 × *geo*2 | → | *real* | **dist** |
| *regions* | → | *real* | **area** |

## VI. SPATIAL QUERYING

Several operations like spatial selection, spatial join and other set operations can be performed on spatial databases.

1.) SELECT operation: It returns the information about objects that match the predicate. For example: suppose there is a region named Sonipat and inside is available operation in algebra then the query for:

a.) "Finding all the cities in Sonipat" will be formulated as

cities select[center inside Sonipat]

b.) "Find cities no more than 100 km from Delhi"(Delhi being a point value)

cities select[dist(center,Delhi)<100]

2.) Join operation: It compares the two objects on their attribute values through predicate. Example:

a.) Combine cities with their states

We use the operation as [cities states join]

3.) Overlay operation is used to compute the elementary regions that result from overlaying of two partitions.

## VII. SPATIAL INDEXING

Spatial indexing method divides the space into manageable number of smaller subspaces, which can be further divided into smaller subspaces and so on. The partitioning continues until the unpartitioned subspace contains the objects that can be stored in a data page. While designing the index structures for spatial databases the storage space must be efficiently utilized and the information retrieval should be fast and easy. Data structures like B trees have been designed for efficient insertion and deletion in databases.

Spatial indexing is used to look up the values that match the predicate in efficient manner. There are two ways to provide spatial indexing:

i.) Dedicated external spatial data structures are added to the system that provide the attributes for spatial databases e.g. a B-tree does for standard attributes, and

ii.) spatial objects are mapped into a one-dimensional space so that they can be stored *within* a standard one dimensional index such as a B-tree.

Approximation technique is used in spatial databases where the object is stored in terms of one or more spatial keys rather than the object itself. The use of approximations help in the query processing as the process is reduced to 2 steps of **filtering** and **refinement**.

Filtering returns the set of objects that are a superset of objects that fulfil predicate. In refinement for each candidate object which is the result of filtering the exact geometry is checked.

A spatial index structure organises the objects in buckets. Each bucket has some rectangular region in which the objects are stored. Figure 2 shows a partition where each bucket region can store a maximum of 3 points.

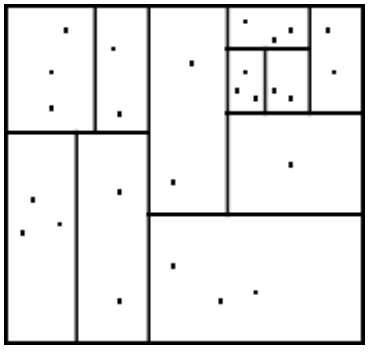

Fig. 2. Kd tree partitioning of 2D space

There are various techniques by which existing indexes are integrated into spatial indexes:

1.) Transformation technique:
In this method the objects with p vertices in q dimensional space are mapped to points in a pq-dimensional space. Example, a 2-dimensional rectangle with the top right corner having coordinates (x1, y1) and bottom left corner having coordinates (x2, y2) is given then it will be represented as a point in 4-dimensional space where each attribute like top, bottom, left and right will take a different dimension.
In this technique the major weakness is that the intersection search is inefficient.

2.) Non overlapping native space technique

a.) An n-dimensional data space is divided into pair wise disjoint subspaces which are then indexed. An object identifier is used to store all the subspaces it intersects.

3.) Overlapping native space technique

A non-zero sized object is included in more than one subspace thereby instead of disjoint subspaces overlapping subspaces in which objects are totally included in only one of subspaces is used. These subspaces are arranged as a hierarchical index.

## VIII. INCONSISTENCY IN SPATIAL DATABASES

Various types of inconsistencies may occur in spatial information like:

1.) Uncertainty: It occurs due to lack of information about the object. For example. If we don't know the distance between NOIDA and AGRA then we cannot calculate the time taken to travel from NOIDA to AGRA.
2.) Incompletion: It refers to the situation where some data values are missing. Example missing road in map.
3.) Inconsistency: It refers to the existence of two contradictory values of a single object. Example NOIDA is 150 km from AGRA and NOIDA is 200 km from AGRA.

## IX.) SPATIAL DATA MINING

Spatial data mining refers to the process of the retrieval of information or patterns that are not explicitly stored in the spatial databases. Spatial data mining methods are used for the better understanding of spatial data, identifying the relationships between spatial data and non- spatial data, query optimization in spatial databases etc.
Statistical Spatial analysis is the most commonly and widely used data mining technique. It assumes that the spatial data are independent which in fact is not true as the spatial data are interrelated with their neighboring objects. Statistical method cannot handle symbolic values and non linear rules and are also very costly in the result computation. Several Machine learning techniques like *learning from examples* and *generalization* and *specialization* are used in spatial data mining.

## X.) SPATIAL DATA MINING ARCHITECTURE

Matheus architecture is the most general and widely used architecture in spatial data mining. This architecture is user controlled. All the predefined information about the objects is stored in the knowledge base which is fetched by the DB interface for query optimization. The information which is useful for the pattern recognition is decided by the Focus Component and fed as input to the pattern extraction. The output is then monitored and evaluated by Evaluation module and duplicate values are removed. All the components interact using the Controller. The data mining architecture is shown in figure 3.
Geographic data consists of the spatial objects and the non spatial information about these objects (which can be stored in database as a pointer to the spatial description of object). Spatial data is characterised by geometric as well as topological characteristics where geometric characteristics involve the information about length, area, perimeter etc and topological characteristics include the information about neighbours, intersection etc.

## XI.) SPATIAL DATA MINING METHODS

Various methods have been designed for mining the data related to geometric space like points, polygon, rectangles, network and other complex objects. There are various kinds of rules associated with spatial data mining.

a.) Characteristic Rule: It refers to the general description of object data. Example rule describing the general price range of shops in various geographic regions of a city.
b.) Discriminant Rule: It refers to the properties or features that distinguish one object from other. Example the comparison of the various shops prices in different regions.
c.) Association Rule: It refers to the association of one object with other.

## XII.) GENERALIZATION BASED MINING

*Learning from examples* when used in combination with generalization technique is used for mining the small databases.It is not suitable for mining large spatial databases due to the complexity of the algorithms as algorithms are exponential in the number of examples and noisy data is not handled by algorithms efficiently.
In generalization based techniques there is a prerequisite of the presence of background knowledge in advance in the form of concept hierarchies which are either generated through data analysis or are provided explicitly by the experts. An example of concept hierarchy for agricultural use is shown as in figure 4. We start moving up from the lower level while generalizing the information like in the given example apple and orange can be generalized to fruits which on moving one level up is generalized to cash crops that also includes vegetable. Similar type of hierarchies and generalizations also exist in spatial data. We perform the attribute-oriented induction on this hierarchy in any of the three ways:

a) When the attribute values are changed to generalized valued then moving up in the hierarchy.
b) When there are many values for an attribute and further generalization is not possible then attribute values are removed.
c) Similar rows are merged together.

And this induction continues until the attribute values in the generalization table reaches the generalization threshold.

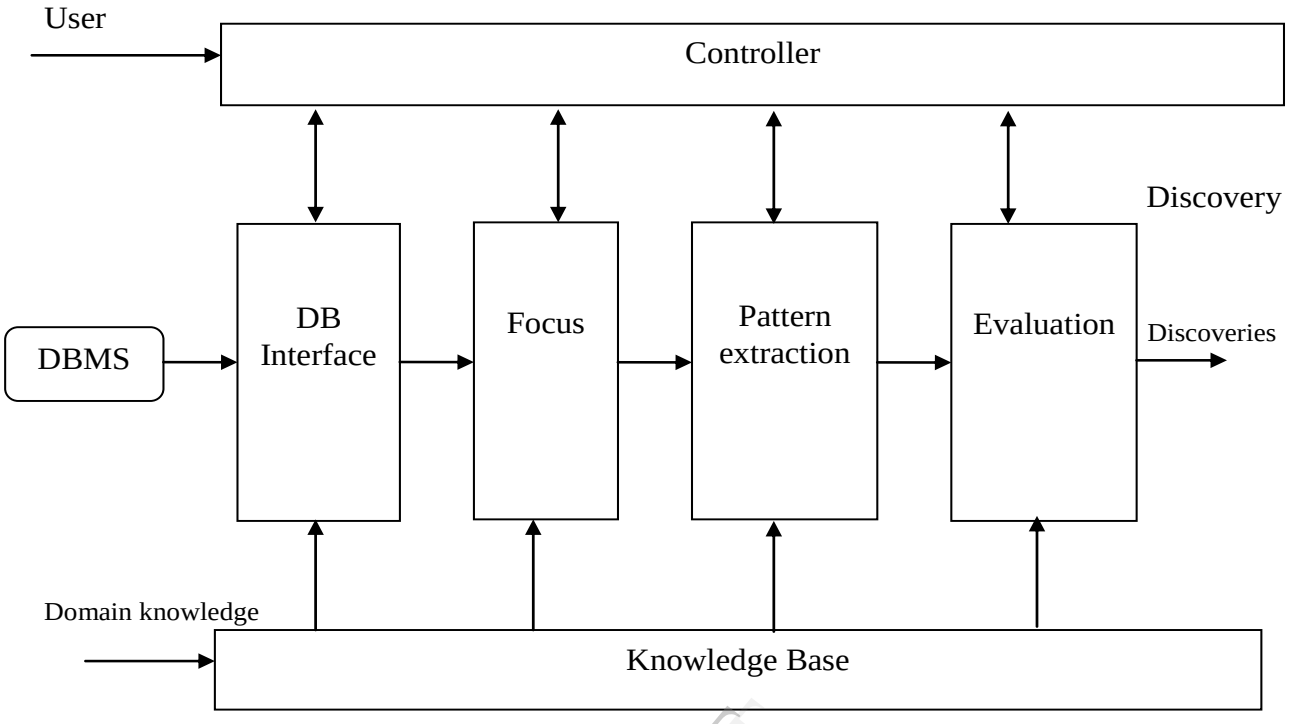


Fig.3. Spatial data mining architecture

## XIII.) ALGORITHMS FOR GENERALIZATION BASED MINING

*Spatial data dominant Algorithm:*

Assumption*:*
Mining process is started by the user by providing a request in the form of SQL query.
Algorithm:
**STEP 1**: background knowledge i.e. data provided in the query is collected.
**STEP 2:** generalization is performed on spatial data until the generalization threshold value is reached by merging the spatial regions according to information in the concept hierarchy.
**STEP 3**: Non spatial data is retrieved and analyzed for each of spatial object using attribute induction.
Complexity:
Complexity of the algorithm is O(MlogM) where M is the number of spatial objects.

*Non-Spatial data dominant Algorithm:*

Assumption*:*
Mining process is started by the user by providing a request in the form of SQL query.
Algorithm:
**STEP 1:** background knowledge i.e. data provided in the query is collected.
**STEP 2:** generalization is performed on non-spatial attributes until the generalization threshold value is reached.
In this, attribute oriented induction is being performed on non-spatial data the pointers to spatial objects are collected as a set and put with the generalized non-spatial data.
**STEP 3:** Neighboring areas with same generalized values are merged together based on the function adjacent_to.
Complexity:
Complexity of the algorithm is O(MlogM) where M is the number of spatial objects.

### WEAKNESSES OF GENERALIZATION BASED ALGORITHMS

1.) When the concept hierarchies do not exist in priori.
2.) In generalization based algorithms the generalization is performed by moving one level upward i.e. the lower levels are merged together to form the higher level regions. The concept hierarchy and information in the levels will depend on the user.

## XIV.) CLUSTERING BASED MINING

Clustering resembles the *unsupervised learning* technique in machine learning in the sense that no prior background knowledge is present. Clustering is defined as the grouping of the database objects into meaningful subclasses. The objects lying in the same cluster have similar characteristics where as the objects lying in different clusters does not resemble each other. It is very difficult to determine the number of clusters in advance due to the large size of spatial databases.

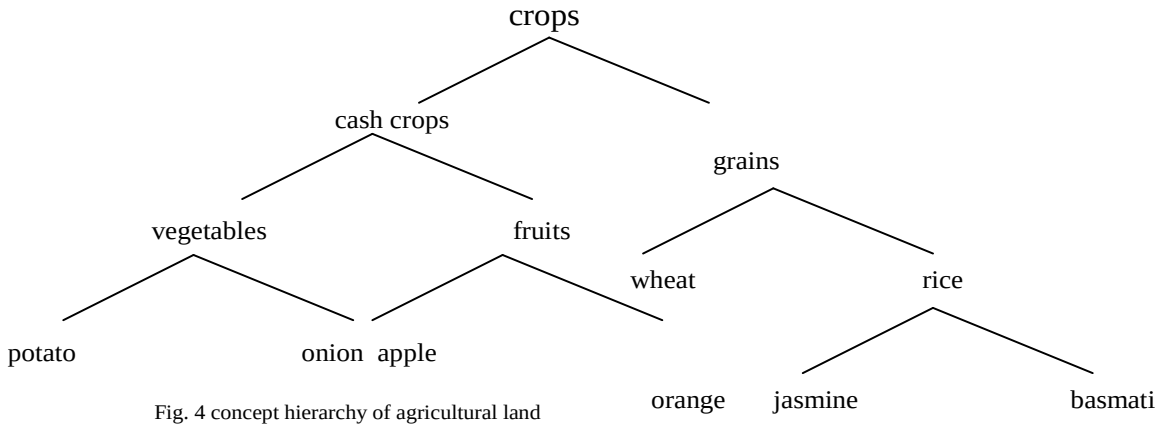


Fig. 4 concept hierarchy of agricultural land

## XV.) DENSITY BASED CLUSTERING ALGORITHMS

*DBSCAN (Density Based Spatial Clustering of Applications with Noise)*

DBSCAN is used to find the clusters of arbitrary shape and with minimal number of input parameters. The points inside the cluster are known as the core points and the points lying at the border of the cluster are called border points.

INPUT PARAMETERS:

1.) Radius of the cluster (Eps)
2.) Minimum points required inside the cluster (Minpts)

TERMS USED:

1.) The Eps neighbourhood of a point p is denoted by $N_{Eps}(p)$ and is defined as:
$N_{Eps}(p) = \{ P \in D | dist(p,q) \leq Eps\}$
2.) A point p is directly density reachable from a point q with respect to Eps, Minpts if:
a.) $p \in N_{Eps}(q)$ and
b.) $|N_{Eps}(q)| >= Minpts$(core point condition)
3.) A point p is density-connected to a point q with respect to Eps and MinPts if there is a point o such that both, p and q are density-reachable from o with respect to Eps and Minpts.

ALGORITHM:

**STEP 1**: An arbitrary point p is selected.
**STEP 2:** All the points that are density reachable with respect to Eps and Minpts from p are identified.
**STEP 3**: If p is a core point, then a cluster is formed and if it is a border point then no point is density reachable from p and hence next point from database is selected as p.
**STEP 4**: Steps 1 to 3 are repeated until all the points in database are processed.

### WEAKNESSES OF DBSCAN ALGORITHM

1.) Applicable only for point objects.
2.) Not suitable in detecting clusters with varied density.

*VDBSCAN (Varied Density Based Spatial Clustering of Applications with Noise)*

The algorithm selects the parameters $Eps_i$ and cluster with varied densities. It calculates the k-dist for each project and partitions the k-dist plots. $Eps_i$ is arbitrarily selected for each density and the database is scanned for each different density.

ALGORITHM:

**STEP 1**: k-dist plot are portioned.
**STEP 2:** Thresholds of parameters $Eps_i$ (i=1,2,3,4…….n )are identified.
**STEP 3**: for each $Eps_i$(i=1,2,3,4…….n )
a.) $Eps=Eps_i$
b.) DBSCAN algorithm is applied for points that are not scanned.
c.) Points are marked as $c_i$.
**STEP 4**: All the marked points are viewed as corresponding clusters.

## XVI.) NON -DENSITY BASED CLUSTERING ALGORITHMS

*Partitioning Around Medoids Algorithm*

Kauffman assumed that there are n objects and PAM finds k clusters by finding a representative object for each cluster called a medoid( centrally located point in a cluster). After selecting k medoids, one for each cluster the algorithm makes a better choice of medoids by analysing each possible pair of object. The best choice of medoid for one iteration is fed in the successive iteration.

Complexity:

Cost of single iteration: $O(k(n-k)^2)$

### DISADVANTAGES OF PAM ALGORITHM

1.) Inefficient for large values of n and k.

*CLARA ALGORITHM (Clustering LARge Applications)*

CLARA algorithm is based on sampling. It does not take the entire data set as input but only a subset of given data and medoid are identified in a similar fashion as in PAM

algorithm. CLARA can deal with large data sets when compared to PAM algorithm.
Complexity:
Complexity of each iteration: $O(kS^2+k(n-k))$ where S is size of sample

DISADVANTAGES OF CLARA ALGORITHM:

1.) If the object O is one of the medoid in best k medoids and for instance if it is not selected during sampling then CLARA will never be able to find best clustering.

*CLARANS ALGORITHM (Clustering Large Applications based on RANdomized Search)*
CLARANS utilizes both PAM and CLARA algorithms as it searches only the subset of data and does not restrict itself to one sample at a particular time.
CLARANS draws a sample with some randomness in each step. If a better neighbor is found CLARANS moves to the neighbor medoid, but the number of neighbors to be tried at each step is fixed by a parameter *maxneighbor*. If no neighbor has a better choice then the algorithm proceeds with current medoid and produces a local optimum. If the local optimum is found then the new node is processed. CLARANS can be treated as similar to searching in graph.

CLARANS is more effective than PAM and CLARA.

DISADVANTAGES OF CLARA ALGORITHM

1.) It makes an assumption that all the objects are stored in main memory but in large databases memory cannot be used to store all the objects.

*SD CLARANS (Spatial data dominant)*
ALGORITHM:
**STEP 1:** In this approach the spatial components of relevant objects are collected and clustered using CLARANS.
**STEP 2:** Attribute oriented induction is performed on non spatial description of objects in each cluster. The result is the high level description of non spatial object in each cluster.

*NSD CLARANS (Non-Spatial data dominant)*
ALGORITHM:
This approach uses the non spatial generalization first and then attribute oriented generalization is performed on non spatial attributes to produce generalized tuples. Then for each generalized tuple spatial components are collected and clustered using CLARANS. SD CLARANS is better in performance when compared to NSD CLARANS.
The limitations of CLARANS algorithm can be removed by the use of spatial access methods like R* tree, CF trees etc.

## XVII.) CONCLUSION

Spatial databases are widely used and the mining of spatial data is a promising field due to the large amount of available spatial data and its wide applicability in fields like remote sensing, traffic management, geographical surveys etc. A large number of algorithms are available for spatial data mining. Few of the existing algorithms along with their advantages and limitations were studied. Spatial data mining methods described assumed the presence of relational databases. The use of object oriented databases instead of traditional databases is also a very challenging and promising field of research.

## XVIII.) REFERENCES

[1] Jiawei Han, Micheline Kamber,"Data Mining Concepts and Techniques."

[2] M. Parimala, Daphne Lopez, N.C. Senthilkumar,"A survey on density based clustering algorithms for mining large spatial databases ".International journal of advanced science and technology vol.31, June, 2011

[3] J. Chimicki, G. Saake, and R. Van Der Meyden, " Logics for Emerging Applications of Databases". Springer-Verlag, 2003.

[4] M. S. Chen, J. Han, P. S. Yu. Data Mining: An Overview from Database perspective.

[5] J. Han, Y Fu, "Exploration of the power of Atttribute oriented induction in data mining."

[6] R. H. Guting," An introduction to Spatial Database Systems" VLDB Journal 3(4):357-400. October 1994

[7] Krzysztof Koperski, Junas AdhikaryJiawei Han, "Spatial Data Mining: Progress and Challenges Survey Paper."

[8] Dongxiang Zhang, Yeow Meng Chee, Anirban Mondal, Anthony K. H. Tung , Masaru Kitsuregawa, "Keyword search in Spatial Databases: Towards Searching by Document". IEEE International Conference on Data Engineering.

[9] "Spatial Databases- Accomplishments and Research Needs" IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, January/Feburary, 1999.

[10] "Query processing in Spatial Databases containing obstacles". International Journal of Geographical Information Science Vol. 19, No. 10, November 2005, 1091-1111.