Review on Spatial Data Mining

[1]MEENAKSHI PANDA, *Gandhi Institute of Excellent Technocrats, Bhubaneswar, India*
[2]SATABDISHREE PANIGRAHI, *Konark Institute of Science and Technology, Bhubaneswar, Odisha, India*

Abstract

Because huge amounts of spatial data have been collected in various applications ranging from remote sensing to geographical information systems (GIS), computer cartography, environmental assessment and planning, and so on, spatial data mining, or mining knowledge from large amounts of spatial data, is a highly demanding field. The amount of data collected significantly outweighed human abilities to interpret it. Data mining has recently expanded its scope beyond relational and transactional databases to include geographical databases. This paper reviews recent work on geographical data mining, including generalization, clustering, and mining spatial association rules, among other topics. It demonstrates that spatial data mining is a promising field with numerous open questions and potential research achievements.

1 Introduction

Large databases have been created as a result of advances in database technologies and data collection techniques such as barcode reading, remote sensing, satellite telemetry, and so on. Data mining or knowledge discovery in databases (KDD) [16, 30, 43] is a potentially emerging field that addresses the need for knowledge/information discovery from data. The finding of interesting, implicit, and previously unknown knowledge from huge datasets is known as database knowledge discovery [20]. Machine learning, database systems, data visualization, statistics, and information theory are all fields that data mining encompasses.

Although data mining has been studied extensively in relational and transaction databases [2, 16, 25, 43], it is also in high demand in other types of databases, such as spatial databases, transactional databases, object-oriented databases, multimedia databases, and so on. The methods of spatial data mining, or the identification of interesting knowledge from spatial data, are the topic of this study.

Spatial data refers to information about items that occupy space. A spatial database is a collection of spatial objects that are represented by spatial data types and spatial relationships.- Among these are ships. Spatial data contains topographical and/or distance information and is frequently organized using spatial indexing structures and accessed using spatial access methods. These inherent characteristics of a spatial database present problems and opportunities for spatial data mining [35]. The extraction of latent knowledge, geographical relations, or other patterns not explicitly stored in spatial databases is referred to as spatial data mining, or knowledge discovery in spatial databases [34].

The foundation for knowledge discovery in databases was built by previous work in machine learning [17, 38, 39], database systems [50, 51], and statistics [9, 19, 31, 47]. Spatial data structures [22, 23, 46], spatial reasoning [10, 12], computational geometry [43], and other breakthroughs in spatial databases prepared the path for the study of spatial data mining. Due to the large amount of spatial data and the complexity of spatial data types and spatial accessing methods, one of the most significant challenges to spatial data mining is the efficiency of spatial data mining algorithms.

Spatial data mining methods can be used to extract useful and consistent information from massive spatial databases. They can be used to understand spatial data, uncover correlations between geographic and noncapital data, build spatial knowledge bases, optimize queries, reorganize data in spatial databases, and capture generic characteristics in a clear and succinct manner, among other things. Geographic Information Systems (GIS), remote sensing, picture database exploration, medical imaging, robot navigation, and other areas where spatial data is employed can all benefit from this. Characteristic and discriminated rules, extraction and description of significant structures or clusters, spatial relationships, and other types of knowledge can be obtained from spatial data. The goal of this survey is to provide a broad perspective of spatial data mining technologies, their strengths and shortcomings, how and when to apply them, and what has been accomplished thus far and what obstacles remain.

Background on Spatial Data Mining

The most frequent method for examining geographic data has been statistical spatial analysis [19, 47]. Because statistical analysis is a well-studied field, there are many methods available, including numerous optimization techniques. It works effectively with numerical data and frequently produces realistic models of spatial phenomena. The assumption of statistical independence among spatially distributed data is a fundamental drawback of this strategy. This poses issues since many spatial facts are in reality interconnected, meaning that spatial objects are influenced by their surroundings. To some extent, regression models incorporating spatially lagged forms of the dependent variables can help solve this problem. Unfortunately, this complicates the modeling process and can lead to errors. Statistical procedures do not work well with data that is incomplete or inconclusive. Another issue with statistical geographic analysis is the high cost of computing the data.

Researchers developed numerous ways for discovering knowledge from massive databases with the advent of data mining. The majority of them focus on relational or transactional databases. These strategies attempted to combine previously developed fields such as machine learning, databases, and statistics. The foundation for geographical data mining was laid by studies like [1, 25, 43]. In geographic data mining, machine learning techniques such as learning from examples, generalization, and specialization are commonly used. It was only a matter of time before the statistical cluster analysis technique was adapted for use in geographical data mining [41]. Other strategies were also applied to knowledge discovery in

1.1.1 Primitives of Spatial Data Mining

Rules: Various kinds of rules can be discovered from databases in general. For example, characteristic rules, discriminate rules, association rules, or deviation and evolution rules can be mined. A spatial characteristic rule is a general description of spatial data. For example, a rule describing the general price range of houses in various geographic regions in a city is a spatial characteristic rule. A spatial discriminate rule is a general description of the features discriminating or contrasting a class of spatial data from other class(es) like the comparison of price ranges of houses in different geographical regions. Finally, a spatial association rule is a rule which describes the implication of one or a set of features by another set of features in spatial databases. For example, a rule associating the price range of the houses with nearby spatial features, like beaches, is a spatial association rule.

Thematic maps show how a single or a few traits are distributed over space. This is in contrast to general or reference maps, which are designed to show the location of items in relation to other spatial objects. Different rules can be discovered using thematic maps. When evaluating the general weather pattern of a geographic region, for example, we could want to look at a temperature themed map. Thematic maps can be represented in two ways: raster and vector. Thematic maps in raster image form have pixels connected with attribute values. For example, the altitude of spatial objects may be coded as the pixel intensity on a map (or the color). A spatial item is represented in a vector representation. Image databases: These are unique geographical databases in which the data is virtually entirely made up of photos or photographs. Remote sensing, medical imaging, and other applications employ image databases. They're commonly saved as grid arrays that indicate image intensity in one or more spectral regions.

1.1.2 Computations, Queries, and Spatial Data Structures

Spatial data mining algorithms employ spatial operations such as spatial joins, map overlays, and nearest neighbour queries, among others. As a result, in spatial data mining, e client spatial access techniques (SAM) and data structures for such computing are also a concern [22]. We'll start with a brief overview of some of the most common spatial data structures and geographical computations.

Data Structures in Space: Points, lines, rectangles, and other shapes make up the spatial data structure. Multidimensional trees have been presented as a way to create indices for these data. Quad trees [46], k-d trees, R-trees, R*-trees, and others are examples. R-tree [23] and its variant R*-tree [6] are two well-known SAMs that have recently received a lot of attention in the literature. Objects saved in the Minimum Bounding Rectangles are used to approximate objects contained in R-trees (MBR). Every node has an R-tree that stores a set of rectangles. Pointers to representations of polygon bounds and polygon MBRs are stored at the leaves. Each rectangle at the internal nodes is linked to a child and represents the smallest bounding rectangle of all the rectangles in the child.

Spatial Computations: Spatial join is one of the most expensive spatial operations. In order to make spatial queries e client spatial join has to be e client as well. Brinkho et al. proposed an e client multilevel processing of spatial joins using R*-Trees and various approximation of spatial objects [8]. The r e s t step let ends possible pairs of intersecting objects using rest their MBRs and later other approximations. In the second step - renitent - detailed geometric procedure is performed to check for intersection. Another important spatial operation, map overlay, is especiallyimportant in Geographic Information Systems.

Spatial Query Processing: Optimization strategies for spatial query processing are outlined in Aref and Samet [5]. The authors proposed an architecture for spatial database called SAND (spatial and non spatial data) architecture, which is a model of the extended relational database with spatial operations [4]. This architecture provides both spatial and non spatial components of spatial database to participate in query processing and optimization.

1.2 Spatial Data Mining Architecture

Various architectures (models) have been proposed for data mining. They include Han's architecture for general data mining prototype DBLEARN/DBMINER [24], Holsheimer et al's parallel architecture [29], and Matheus et al.'s multi component architecture [37]. Al- most all of these architectures have been used or ex- tended to handle spatial data mining. Matheus et al.'s architecture seems to be very general and has been used by other researchers in spatial data mining, including Ester et al. [13]. This architecture - comparable to oth- ers - is presented in Figure 1. In this architecture, the user may control every step of the mining process. Back- ground knowledge, like spatial and non-spatial concept hierarchies, or information about database, is stored in a knowledge base. Data is fetched from the storage us- ing the DB interface which enables optimization of the queries. Spatial data index structures, like R-trees, may be used for e cient processing. The Focusing Compo- nent decides which parts of data are useful for patternrecognition. For example, it may decide that only some attributes are relevant to the knowledge discovery task, or it may extract objects whose usage promises good results. Rules and patterns are discovered by the Pat- tern Extraction module. This module may use statistical, machine learning, and data mining techniques in conjunction with computational geometry algorithms to perform the task of finding rules and relations. The in- terestingness and significance of these patterns is then processed by Evaluation module to possibly eliminate obvious and redundant knowledge. The four last com- ponents may interact between themselves through the Controller part.

1.3 Organization of the paper

The rest of the paper is organized as follows. In Section 2 we survey the methods for spatial data mining. We categorize the methods and discuss each in detail. Section 2.1 describes generalization based methods, Section 2.2 discusses clustering based methods, Section 2.3 presents the methods used to explore spatial associations, Section 2.4 describes pattern recognition meth- ods, and nally in Section 2.5 other interesting methods are outlined. We present suggestions and future directions in Section 3, and we conclude our discussion in Section 4.

2 Methods for Knowledge Discovery in Spatial Databases

Geographic data consist of spatial objects and non- spatial description of these objects. Non-spatial description of spatial objects can be stored in a traditional relational database where one attribute is a pointer to spatial description of the object [4]. Spatial data can be described using two different properties, geometric and topological. For example, geometric properties can be spatial location, area, perimeter, etc., whereas topological properties can be adjacency (object A is neighbor of object B), inclusion (object A is inside in object B), and others. Thus, the methods for discovering knowledge can be focused on the non-spatial and/or spatial properties of spatial objects.

The algorithms for spatial data mining include gene- realization-based methods for mining spatial characteristic and discriminate rules [25, 35, 41], two-step spatial computation technique for mining spatial association rules [34], aggregate proximity technique for ending characteristics of spatial clusters [33], etc. In the following sections, we categorize and describe a numberof these algorithms.

2.1 Generalization-Based Knowledge Discovery

One of the widely used techniques in machine learning is learning from examples [38]. This method is often combined with generalization [39]. This approach cannot be directly adopted for large spatial databases because: 1) the algorithms are exponential in the number of examples, and 2) it does not handle noise and inconsistent data very well. Han et al. [25] modified these techniques and gave an attribute-oriented (as opposed to the tuple-oriented in machine learning algorithms) induction algorithm to mine knowledge from large relational databases. Later Lu et al. [35] extended this technique to spatial databases. Thus, theassumptions that are made for relational databases are also carried to spatial data mining.

The generalization-based knowledge discovery re- quires the existence of background knowledge in the

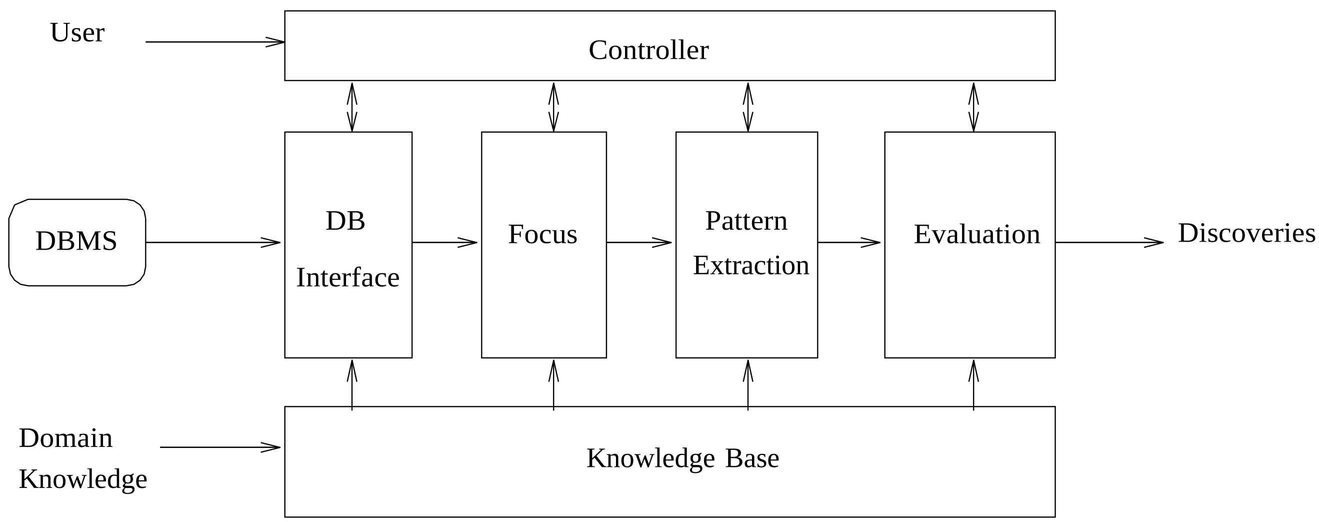

Figure 1: An architecture for a KDD system

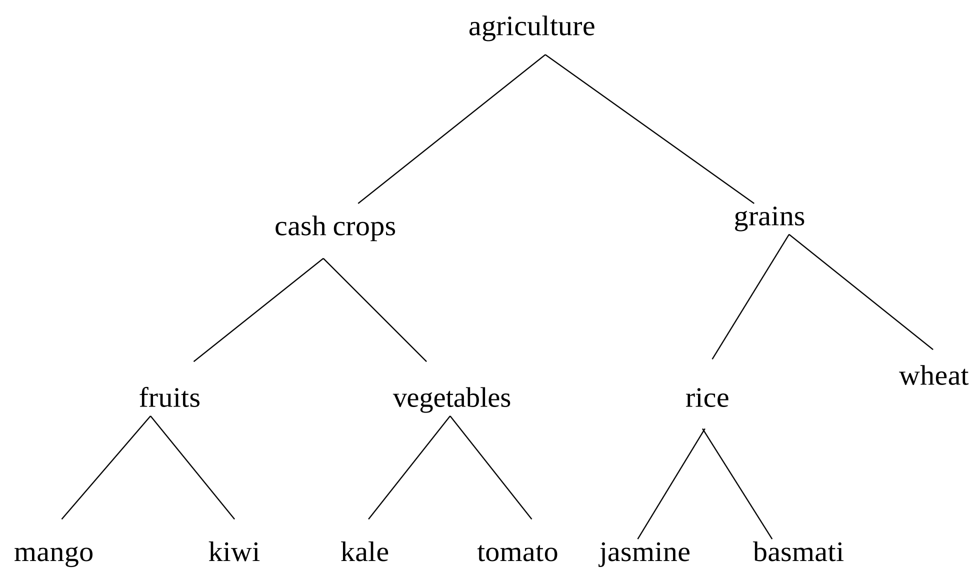

Figure 2: Example of agricultural land use concept hierarchy

form of concept hierarchies. In the case of spatial databases, there can be two kinds of concept hierar- chies, non-spatial and spatial. Concept hierarchies can be explicitly given by the experts, or in some cases they can be generated automatically by data analysis [26]. An example of a concept hierarchy for agricultural land use is shown in Figure 2. As we ascend the concept tree, information becomes more and more general, but still remains consistent with the lower concept levels. For example, in Figure 2 both jasmine and basmati can be generalized to the concept rice which in turn can be generalized to concept grains, which also includes wheat. A similar hierarchy may exist for spatial data. For ex- ample, in a generalization process, regions representing counties can be merged to provinces and provinces can be merged to larger regions. Attribute-oriented induc- tion is performed by climbing the generalization hier- archies and summarizing the general relationships be- tween spatial and non-spatial data at higher concept levels. It can be done on non-spatial data by (a) climb- ing the concept hierarchy when attribute values in a tuple are changed to the generalized values, (b) remov- ing attributes when further generalization is impossible and there are too many di erent values for an attribute, and (c) merging identical tuples. Induction is continued until every attribute is generalized to the desired level. The desired level is reached when the number of dif- ferent values for the attribute in the generalized table is no greater than the generalization threshold for this attribute. During the process of merging of identical tu- ples the number of merged tuples is stored in additional attribute count to enable quantitative presentation of acquired knowledge. Lu et al. [35] presented two gener- alization based algorithms, spatial-data-dominant and non-spatial-data-dominant generalizations. Both algo- rithms assume that the rules to be mined are general data characteristics and that the discovery process is initiated by the user who provides a learning request (query) explicitly, in a syntax similar to SQL. We will brie y describe both algorithms as follows:

Spatial-Data-Dominant Generalization: In the

rst step all data described in the query are collected. Given the spatial data hierarchy, generalization can be performed rst on the spatial data by merging the spatial regions according to the description stored in the concept hierarchy. Generalization of the spatial objects continues until the spatial generalization threshold is reached. The spatial generalization threshold is reached when the number of regions is no greater than the threshold value. After the spatial-oriented induction process, non-spatial data are retrieved and analyzed for each of the spatial objects using the attribute-oriented induction technique as described above. An example of a query and the result of the execution of the spatial-data-dominant generalization algorithm is presented in Figure 3. In this example, temperature in the range [20, 27) is generalized to moderate, and temperature in the range [27, 1) to hot. The answer to the query is the description of all regions using a disjunction of a few predicates which characterize each of the generalized regions. Temperature measured in the east-central region of British Columbia is in the range [22, 30]. Thus, in our example, the description of the temperature weather pattern in this region is hot or moderate. The computational complexity of the algorithm is O(N logN), where N is the number of spatial objects.

Non-spatial-Data-Dominant Generalization:
This method also starts with collecting all data relevant to the user query. In the example presented in Figure 4 the DB interface extracts the precipitation data relevant to the province and time period speci ed in the query. In the second step the algorithm performs attribute- oriented induction on the non-spatial attributes, gener- alizing them to a higher (more general) concept level. For example, the precipitation value in the range (10 in., 15 in.] can be generalized to the concept wet. The generalization threshold is used to determine whether to continue or stop the generalization process. In this step the pointers to spatial objects are collected as a set and put with the generalized non-spatial data. In the third and the last step of the algorithm, neighboring areas with the same generalized attributes are merged together based on the spatial function adjacent to. For example, if in one area the precipitation value was 17 in., and in neighboring area it was 18 in. both precipita- tion values are generalized to the concept very wet and both areas are merged. Approximation can be used to ignore small regions with di erent non-spatial descrip- tion. For example, if the majority of area land can be described as industrial, but a few gas stations exist in this area the whole area can be described as industrial one. The result of the query may be presented in the form of a map with a small number of regions with high level descriptions as it is shown in Figure 4. The compu-tational complexity of this algorithm is also O(N logN),where N is the number of spatial objects.

We presented two generalization based algorithms that assumed the concept hierarchies to be given or generated automatically. However, as pointed out before, there may be cases where such hierarchies are not present a priori. Another problem with previous algorithms is that the spatial components of the databases are explored by merging regions at lower levels of the concept hierarchy to form region(s) at higher levels of the hierarchy. Both of these facts suggest that the quality and the interestingness of the mined characteristic rules is going to be much dependent upon the given concept hierarchy(ies). In many cases such

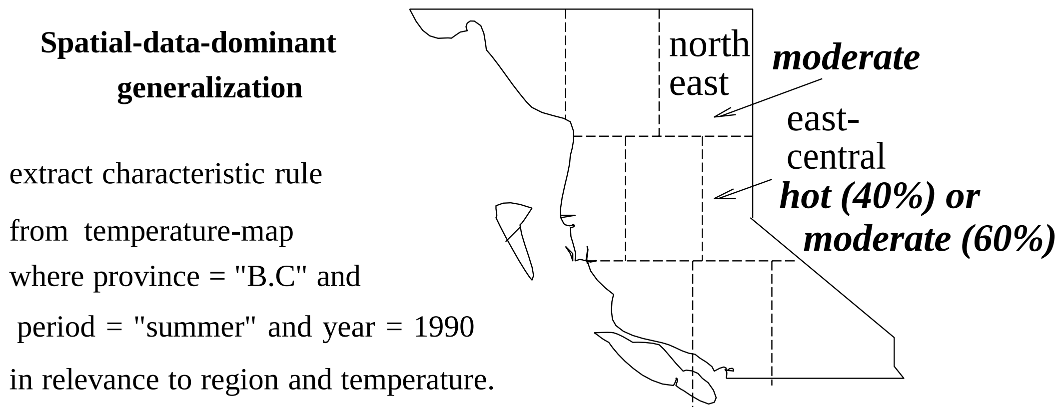

Figure 3: Example of a query and the result of the execution of the spatial-data-dominant generalization method

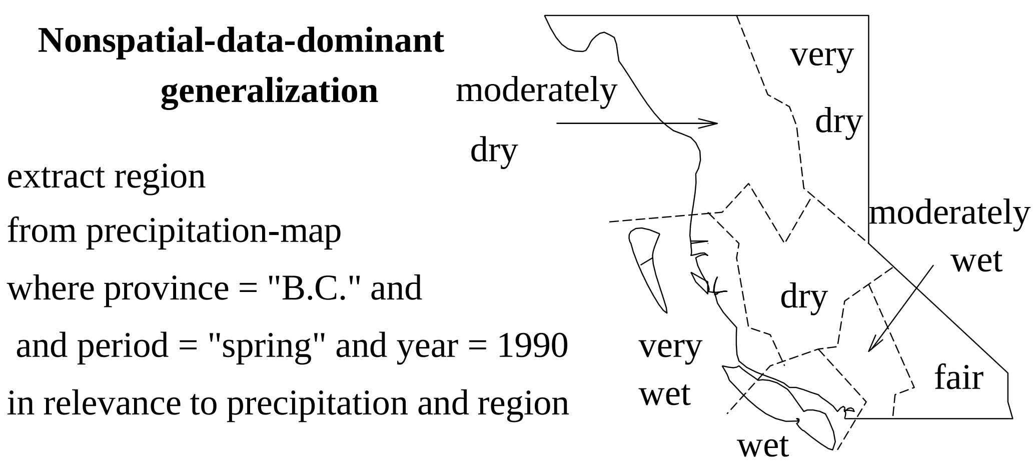

Figure 4: Example of a query and the result of the execution of the non-spatial-data-dominant generalization method

hierarchies are given by the experts and they may benot entirely appropriate. Therefore, we would like to nd algorithms that do not need to use these hierarchies. We will describe an algorithm not depending on spatial concept hierarchies in the next section.

2.2 Methods Using Clustering

Cluster analysis is a branch of statistics that has been studied extensively for many years. The main advantage of using this technique is that interesting structures or clusters can be found directly from the data without using any background knowledge, like concept hierarchies. A similar approach in machine learning is known as unsupervised learning. We can exploit the results of research on clustering techniques in the spatial data mining process as proposed in [41].

Clustering algorithms used in statistics, like PAM or CLARA [31], are reported to be ine cient from the computational complexity point of view. As for the e ciency concern, a new algorithm, called CLARANS (Clustering large Applications based upon RANdom- ized Search), was developed for cluster analysis. Ex- perimental evidence showed that CLARANS outper- forms the two existing cluster analysis algorithms, PAM (Partitioning Around Medoids) and CLARA (Cluster- ing LARge Applications). Ng and Han used CLARANS in spatial data mining algorithms, SD(CLARANS) and NSD(CLARANS). First, we will brie y describe the three cluster analysis algorithms.

The PAM algorithm was developed by Kaufman and Rousseeuw [31]. Assuming that there are n objects, PAM nds k clusters by rst nding a representative object for each cluster. Such a representative, which is the most centrally located point in a cluster, is called a medoid. After selecting k medoids, the algorithm repeatedly tries to make a better choice of medoids analyzing all possible pairs of objects such that one object is a medoid and the other is not. The measure of clustering quality is calculated for each such combination. The best choice of points in one iteration is chosen as the medoids for the next iteration. The cost of a single iteration is O($(k\,n \quad k)^2$). It is therefore computationally quite ine cient for large values of n and k.

The CLARA algorithm was proposed by Kaufman and Rousseeuw [31] as well. The di erence between the PAM and CLARA algorithms is that the latter one is based upon sampling. Only a small portion of the real data is chosen as a representative of the data and medoids are chosen from this sample using PAM. The idea is that if the sample is selected in a fairly random manner, then it correctly represents the whole data set and therefore, the representative objects (medoids) chosen, will be similar as if chosen from the whole data set. CLARA draws multiple samples and outputs the best clustering out of these samples. As expected, CLARA can deal with larger data sets than PAM. The complexity of each iteration now becomes O(kS^2 +k(n k)), where S is the size of the sample. The authors indicated through their experimental results that samples of size 40+2k give good results.

It is easy to realize that PAM searches for the best k medoids among a given data set whereas CLARA searches for the best k medoids among the selected sample of the data set. Let us suppose that object O_i is one of the medoids in the best k medoids. Thus, if during sampling O_i is not selected, then CLARA will never nd the best clustering. This is exactly the tradeo for e ciency. Ng and Han's [41] proposed CLARANS algorithm which tries to mix both PAM and CLARA by searching only the subset of the data set and it does not con ne itself to any sample at any giventime. While CLARA has a xed sample at every stage of the search, CLARANS draws a sample with some randomness in each step of the search. The clustering process can be presented as searching a graph where every node is a potential solution, i.e., a set of k medoids. The clustering obtained after replacing a single medoid is called the neighbor of the current clustering. The number of neighbors to be randomly tried is restrictedby the parameter maxneighbor. If a better neighbor is found CLARANS moves to the neighbor's node and theprocess is started again, otherwise the current clustering produces a local optimum. If the local optimum is found CLARANS starts with new randomly selected node in search for a new local optimum. The number of local optima to be searched is also bounded by the parameter numlocal. CLARANS has been experimentally shown to be more e cient than both PAM and CLARA. The authors claim that the computational complexity of every iteration in CLARANS is basically linearly proportional to the number of objects. This claim has been supported by Ester et al. in [13]. It should be mentioned that CLARANS can be used to nd the most natural number of clusters k_{nat}. The authors adopted a heuristic of determining k_{nat}, which uses silhouette coe cients[1], introduced by Kaufman and Rousseeuw [31]. CLARANS also enables the detection of outliers, e.g., points that do not belong to any cluster.

Based upon CLARANS, two spatial data mining algorithms were developed in a fashion similar to the algorithms discussed earlier in this section: spatial dominant approach, SD(CLARANS) and non-spatial dominant approach, NSD(CLARANS). Both algorithms assume that the user speci es the type of the rule to be mined and relevant data through a learning request ina similar way as in the experimental database mining prototype, DBLearn [25].

[1] It is a property of an object that speci es how much the objecttruly belongs to the cluster.

Algorithm SD(CLARANS)

In this spatial dominant approach, spatial compo- nent(s) of the relevant data items are collected and clus- tered using CLARANS. Then, the algorithm performs an attribute-oriented induction on non-spatial descrip- tion of objects in each cluster. The result of the query presents high-level non-spatial description of objects in every cluster. For example, one can nd that in Vancou- ver expensive housing units are clustered in 3 clusters. In the downtown cluster there are mainly expensive con- dominiums; in the waterfront cluster mansions and sin- gle houses are located; and the third cluster consists mainly of single houses.

Algorithm NSD(CLARANS)

This non-spatial dominant approach rst applies non- spatial generalizations. Attribute-oriented generaliza- tion is performed on the non-spatial attributes and pro- duces a number of generalized tuples. For example, the descriptions of expensive housing units can be gener- alized to single houses, mansions and condominiums. Then, for each such generalized tuple, all spatial com- ponents are collected and clustered using CLARANS to nd k_{nat} clusters. In the nal step, the clusters obtained that way are checked to see if they overlap with clus- ters describing other types of objects. If so, then the clusters are merged, and the corresponding generalized non-spatial descriptions of tuples are merged as well.

Depending upon the rules or the form of knowl- edge that user wants to discover, it may be betterto choose one or the other of the above two algo- rithms. Usually SD(CLARANS) is more e cient than NSD(CLARANS). But, when the distribution of points is mainly determined by their non-spatial attributes NSD(CLARANS) may have an edge.

CLARANS in large Spatial Databases Focusing

Methods

Ester et al. [13] pointed out some of the drawbacks of the CLARANS clustering algorithm [41]. First of all, CLARANS assumes that the objects to be clustered are all stored in main memory. This assumption may not be valid for large databases and that is why disk-based methods could be required. Secondly, the e ciency of the algorithm can be substantially improved by modifying the focusing component of the algorithm (see architecture in Figure 1).

The rst drawback is alleviated by integrating CLA- RANS with e client spatial access methods, like R- tree. R*-tree supports the focusing techniques that Es- ter et al. proposed to reduce the cost of computations. It showed that the most computationally expensive step* of CLARANS is calculating the total distances between the two clusterings. Thus, the authors proposed two approaches to reduce the cost of this step.

The rst one is to reduce the number of objects to consider. A centroid query returns the most central object of a leaf node of the R*-tree where neighboring points are stored. Only these objects are used to compute the medoids of the clusters. Thus, the numberof objects taken for consideration is reduced. This technique is called focusing on representative objects. The drawback is that some objects, which may be better medoids, are not considered, but the sample is drawn in the way which still ensures good quality of clustering.

The other technique to reduce the computations is to restrict the access to certain objects that do not actually contribute to the computation. The authors further gave two di erent focusing techniques which try to exploit this approach: focus on relevant clusters, and focus on a cluster. Using R*-tree structure the authors proposed a way of performing computation only on pairs of objects that can improve the quality of clustering instead of checking all pairs of objects as it is done in CLARANS algorithm.

Ester et al. applied the focusing on representative objects to a large protein database to nd the segmen- tation of protein surfaces so as to facilitate the so-called docking queries. They reported that when the focusing technique was used the e ectiveness decreased just from 1.5% to 3.2% whereas the e ciency increased by factor 50, which was the number of points stored in a disk page. The measure of e ectiveness used is the average distance of the resulting clustering whereas the measure of e ciency used is the CPU time.

Clustering Features and CF trees

R-trees are not always available and their construction may be time consuming. Zhang et. al. [52] pre- sensed another algorithm - BIRCH (Balanced Iterative Reducing and Clustering) - for clustering of large sets ofpoints. The method they presented is the incremental one with possibility of adjustment of memory require- ments to the size of memory that is available. The au- thors used concepts called Clustering Feature and CF tree.

A Clustering Feature CF is the triple summarizing information about sub clusters of points. Given N d-dimensional points in the sub cluster: $\{X_i\}$, CF is de ned as

$$CF = (N; \vec{LS}; SS)$$

where N is the number of points in the subcluster, $\vec{LS}$ is the linear sum on N points, i.e., $\sum_{i=1}^{N} \vec{X_i}$, and SS is the square sum of data points, i.e., $\sum_{i=1}^{N} \vec{X_i}^2$. The Clustering Features are su cient for computing clusters and they constitute an e cient information storage method as they summarize information about the subclusters of points instead of storing all points.

A CF tree is a balanced tree with two parameters: branching factor B and threshold T. The branching factor speci es maximum number of children. The threshold parameter speci es the maximum diameter of subclusters stored at the leaf nodes. By changing the threshold value we can change the size of the tree. The non-leaf nodes store sums of their children's CFs, and thus, they summarize the information about their children. The CF tree is build dynamically as data points are inserted. Thus, the method is an incremental one. A point is inserted to the closest leaf entry (subcluster). If the diameter of the subcluster stored in the leaf node after insertion is larger than the threshold value, then, the leaf node and possibly other nodes are split. After the insertion of the new point the information about it is passed towards the root of the tree. One can change the size of the CF tree by changingthe threshold. If the size of the memory that is needed for storing the CF tree is larger than the size of the mainmemory, then a larger value of threshold is speci ed and the CF tree is rebuilt. The rebuild process is performed by building a new tree from the leaf nodes of the old tree.Thus, the process of rebuilding the tree is done without the necessity of reading all the points. Therefore, for building the tree data has to be read just once. The authors present also some heuristics for dealing with outliers and methods for improving the quality of CF trees by additional scans of the data.

Zhang et. al. claim that any clustering algorithm, including CLARANS may be used with CF trees. The CPU and I/O costs of the BIRCH algorithm are O(N). The authors performed a number of experiments which showed linear scalability of the algorithm with respectto number of points, insensibility to the input order, and good quality of clustering of the data.

2.3 Methods Exploring Spatial Associations

All methods that we discussed in previous sections nd only characteristic rules that characterize spatial objects according to their nonspatial attributes. In many situations we want to discover spatial association rules, rules that associate one or more spatial objects with other spatial objects. The concept of association rules was introduced by Agrawal et al. [1] in a study of mining large transaction databases. Koperski and Han [34] extended this concept to spatial databases. A spatial association rule is of the form X ! Y (c%), where X and Y are sets of spatial or nonspatial predicates and c% is the con dence of the rule. For example, the following rule is a spatial association rule: is a(x,school) ! close to(x,park) (80%). This rule states that 80% of schools are close to parks. There are various kinds of spatial predicates that could constitute a spatial association rule. Some examples are: topological relations like intersects, overlap, disjoint, etc.; spatial orientations like left of, west of, etc.; distance information, such as close to, far away, etc.

To con ne the number of discovered rules, the con- cepts of minimum support and minimum con dence are used. The intuition behind this is that in large databases, there may exist a large number of associa- tions between objects but most of them will be applica-ble to only a small number of objects, or the con denceof rules may be low. For example, the user may not beinterested in the relation associating 5% of houses anda single school. He/she may be interested in rules that apply to at least 50% of houses. We would like to lter out associations describing small percentage of objects using the minimum support thresholds. We also want to lter out rules with low con dence using minimum con dence threshold. These thresholds can be di erent at each level of non-spatial description of objects since the same thresholds may not nd interesting associa- tions at the lower concept levels where the number of objects having the same description is smaller. Thus, at the lower levels of non-spatial hierarchies the percent- age of objects may not reach the support threshold forthe higher levels [2]. Informally, we can de ne the sup- port of a pattern A in a set S[3] to be the likelihood of the occurrence of pattern A in S, and the con dence of rule X ! Y to be the likelihood that the pattern Y for object O_s occurs whenever X occurs for the same object. A set of predicates P is large in set S at levell of the non-spatial concept hierarchy if the support ofP is no less than its minimum support threshold 0 for level l (it is true for large number of objects), and all ancestors of P from the concept hierarchy are large at their corresponding levels. A strong rule is a rule with large support, i.e., no less than the minimum support threshold, and large con dence, i.e., no less than the minimum con dence threshold. A top-down, progres- sive deepening search method for mining strong spatial association rules is described in [34].

To minimize the number of costly spatial computa- tions a novel two step spatial computation technique for optimization during the search for associations was in- troduced [34]. Computation starts at the high level of spatial predicates like g close to (generalized close to). A pair of objects satis es the predicate g close to if their Minimum Bounding Rectangles are located in the dis- tance no greater then the threshold for this predicate. Thus, we deal with the problem of the intersections of isothetic rectangles. E cient spatial computation algo- rithms and structures like R-trees or plane-sweep tech- niques can be used in this step. More detailed and ner,

[2] See Han and Fu [27] for detailed discussion on the rationale behind the multiple level thresholds for mining multiple level association rules in large transaction databases.

[3] S is the set of objects that are described.

but more expensive, spatial computations are applied at lower concept levels only to those patterns that are large at the level of the predicate g close to. The ra- tionale behind this is that if a pattern is not large atg close to level it certainly will not be large at the level of detailed spatial relations. Filtration of large patterns saves a great deal of computations since there are much fewer spatial association relationships left at the lower concept levels. The ltration process is done using min- imum support at the high levels.

Algorithm for Multiple Level Spatial Association Rules

The mining process is started by a query which is to describe a class of objects S using other task relevant classes of objects, and a set of relevant relations. For example, a user may want to describe parks by pre- senting the description of relations between parks andother objects like: railways, restaurants, zoos, hydro- logical objects, recreational objects, and roads. Fur- thermore, the user can state that he/she is interested only in objects in the distance less than one kilome- ter from a park. The rst step of the algorithm col- lects the task-relevant data. Then, some e cient spa- tial computations are performed as mentioned above to extract spatial associations at the level of generalized spatial relations. These e cient computations look for objects whose minimal bounding rectangles are located in the distance no greater than the threshold to sat- isfy the close to predicate. In this way, objects satis- fying the predicate g close to (generalized close to) arefound. This predicate encompasses exact spatial pred-icates like adjacent to, intersects, distance less than x. The g close to predicates are stored in an extended re- lational database Coarse predicate DB. Every row of the Coarse predicate DB is a description of a single objectfrom the class of objects being described. Description consists of objects which satisfy task relevant predicates. For example, a row related to Stanley Park in Vancou-ver may include restaurant, zoo, main road, inlet, lake and other objects located inside the park or close to it. Each predicate in Coarse predicate DB is checked with the threshold for the top level to lter out task-relevantclasses of objects in the g close to predicates which donot promise getting large predicates. For example, if only 5% of objects from class S satisfy the predicateg close to(s, zoo) and the minimum support threshold on the top level is 15% then the predicates g close to(s, zoo) will be deleted. This ltration results in a database of large predicates (Large Coarse predicate DB). A spa- tial association rules at the coarse level can be gener- ated from Large Coarse predicate DB. This database is further processed using ner spatial computations to produce Fine predicate DB. In the Fine predicate DB, generalized predicates like g close to are changed into exact spatial predicates like adjacent to, intersects, or distance less than x. We call a single predicate, like close to(x, lake), a 1-predicate. The conjunction of k such predicates is called a k-predicate. For example, the predicate close to(x, lake) ^ close to(x, restaurant) is a 2-predicate. This predicate states that the object x is both close to a lake and close to a restaurant The Fine predicate DB is used to produce large k-predicates and generate association rules at multiple concept levels. At each concept level, the algorithm starts with large 1-predicates and iteratively generates large k-predicates until no large (k+1)-predicate can be found by adding a large 1-predicate to any large k-predicate. The algorithm nds large predicates by counting the number of occur-rences of predicates in the database and comparing this number with the support threshold. The predicates and the number of their occurrences in Fine predicate DB are stored in the predicate table. Based on the informa-tion stored in the predicate table the algorithm derives strong rules. For example, if the predicate close to(x, lake) occurs in 100 rows of Fine predicate DB, the predi- cate close to(x, restaurant) occurs in 90 rows, and both predicates close to(x, lake) and close to(x, restaurant) occur together in 80 rows, then the rule \is a(x, park) ^ close to(x, lake) ! close to(x, restaurant) (80%)" may be derived. After nding large predicates on high lev- els of concept hierarchies, the algorithm tries to nd large predicates and rules on lower levels. For example, restaurants may be specialized into oriental restaurants and continental restaurants, and the algorithm may nd relations between parks and these types of restaurants.

The computational complexity of the algorithm is $O(C_c\ n_c + C_f\ n_f + C_{nonspatial})$ [34], where C_c and C_f are average costs of computing each spatial predicate at a coarse and ne resolution level respectively, n_c is the number of predicates that are coarsely computed, n_f is the number of predicates that are nely computed, and $C_{nonspatial}$ is the total cost of generating rules from the predicate databases. It is observed that n_f is smallerthan n_c, but C_c is more e cient that C_f.

The above algorithm, especially the two-step compu-tation technique, is a novel approach towards mining spatial association rules at multiple levels. It requires background knowledge in the form of concept hierar- chies and expects a user to describe the form of the rule s/he wants by giving such information in the mining query. It may be a good idea to work towards integra-tion of this technique with clustering methods to avoidthe necessity of the user having to provide the concepthierarchies for spatial and nonspatial attributes.

2.4 Using Approximation and Aggregation

We discussed a clustering algorithm CLARANS in Section 2.2. The algorithm is an e ective and e cient method of nding where the clusters in the spatial database are, i.e., partitioning data into clusters. However, perhaps the more interesting result would be to nd out why the clusters are there. Knorr and Ng in [33] presented a study motivated by this question. This question can be rephrased as \what are the characteristics of the clusters in terms of the featuresthat are close to them". The problem is how to measure the aggregate proximity, because statements like 90% of the houses in a cluster are close to the feature F are more informative and interesting than statements like one house is close to a certain feature F. The aggregate proximity is the measure of closeness of the set of points in the cluster to a feature as opposed to the distance between a cluster boundary and the boundary of a feature.

One may ask why the authors are not simply using the k nearest neighbor searches using structures like k-d trees, R-trees and its variants, Voronoi diagrams[4], etc. It turns out that such structures are unable to perform the search needed for their purpose. For example, the distance between the cluster and a feature is measured as the distance between the boundaries, not between the points, like centroids. Furthermore, the costs of building and maintaining the indices are prohibitive given the fact that such indices may not be used frequently. Therefore, the authors propose the use of computational geometry concepts [44] to nd out the characteristics of a given cluster in terms of the features close to it. The authors described the algorithm CRH (where C is for encompassing circle, R for isothetic rectangle, and H for convex hull[5]) which uses such concepts as lters to reduce the candidate features at multiple levels. In short, they collect a large number of features from multiple maps and feed them along with the cluster to the algorithm CRH and discover knowledge about spatial relationships as shown in Figure 5.

Algorithm CRH

Knorr and Ng evaluated various computational ge- ometry algorithms for distance computation, and shape descriptions and overlap computations. Taking into ac- count the problem of data distribution in a cluster and various sizes and shapes of the features, the authors chose a technique for computing the distance between a cluster point and feature boundary. For the shape de- scription, the authors chosed minimum bounding struc- tures. They used these structures to develop a multiple ltering approach, with the lters set up in an increas- ing order of accuracy but decreasing order of e ciency.

[4] Voronoi diagram of a set of points S is a set of points having more that one nearest neighbor from the set S.

[5] Convex hull is the minimal, simple closed curve of a set of points such that a line connecting any two points of the set alwayslies on the interior of the boundary of convex hull.

That is, lters that are applied earlier are more e cient but coarser than the later ones.

The algorithm CRH rst applies the encompassing circle lter to the large number of features. Features that are the most promising ones are passed to the isothetic rectangle lter. These two lters eliminate a large number of features and only a small number of features is passed to the nal convex hull lter. Then, the CRH algorithm calculates the aggregate proximity of points in the cluster to the convex boundary of each feature, upon which the features are ranked. Also, each lter has its own threshold, which is the minimum number of features to pass on to the next lter. When the number of features found lies below the threshold, the cluster is enlarged to encompass more features to pass the threshold limit. Shape enlargement can be achieved by the linear policy (enlarge the shape by constant distance), or by the bisection policy. Bisection policy performs enlargement or diminution of the area by a distance which decreases logarithmically. This policy checks if enough features are in the area of shape and enlarges or decreases the area according to the need. The problem with this method is that a feature may have to be tested for overlap with a cluster many times. The technique, which is called by the authors mem-oization, can be used to avoid multiple computations by storing the distance between each feature and the cluster the rst time the intersection test is performed. Depending upon the shapes, circles, rectangles or con- vex hulls, the minimum distance between the circumfer- ences, the boundaries of the rectangles, or the bound-aries of the polygons respectively are stored. Finally, the algorithm reports the features with the smallest ag- gregate proximities showing minimum and maximum distances of points in the cluster to the feature, aver-age distance, and percentages of points located in the distance less than speci ed thresholds.

The algorithm CRH is experimentally reported to have the response time of less than two seconds for processing 50,000 features. Furthermore, it is empirically shown to be scalable and the memoization policy is found to be the most consistent and e cient of all the shape enlargement policies.

2.5 Mining in Image Databases

Knowledge mining from Image Databases can be viewed as a case of spatial data mining. There have been stud- ies, led by Fayyad et al. [14, 15, 48], on the automatic recognition and categorization of astronomical objects. The authors presented a system [15] for identifying vol- canos on the surface of Venus from images transmitted by the Magellan spacecraft. The Magellan transmit- ted more than 30,000 high resolution synthetic aper- ture radar images of the surface of Venus from di erent angles. The system is composed of three basic compo-

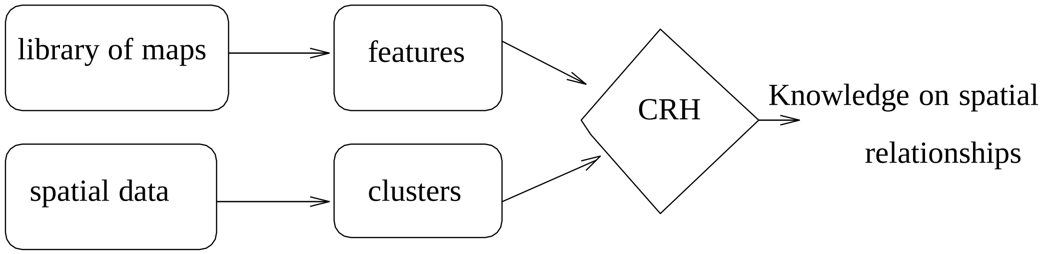

Figure 5: Using CRH for knowledge discovery in spatial databases

nents: data focusing, feature extraction, and classi ca- tion learning. Like all other data focusing techniques, the rst component increases the overall e ciency of the system by rst identifying the portion of the image being analyzed that is most likely to contain a volcano. This is achieved by comparing the intensity of the cen- tral pixel of a region to the estimated mean background intensity of its neighborhood pixels. The second com- ponent of the system extracts interesting features from the data. Standard methods used in pattern recogni- tion like edge detection or Hough transform, deal poorly with the variability and noise presented in the case of natural data. Since it is di cult to nd attributes de- scribing volcanos exactly, matrices containing volcanos images were decomposed to eigenvectors. Eigenvalues were treated as attributes describing volcanos. Then the nal task, which is performed by the rest of the system, is to discriminate between volcanos and other objects looking like volcanos. Such \false alarms" are caused by objects on the surface of Venus causing inten- sity deviations [7]. The nal component of the system uses training examples provided by the experts to cre- ate a classi er that can discriminate between volcanos and \false alarms". The decision tree method [45] was used for this task. The incidence angle of the synthetic aperture radar to the planet instrument strongly in u- enced images of volcanos. Thus, the images were nor- malized according to this angle. The obtained accuracy was about 80%.

In general, it is di cult for experts to provide classi - cations with 100% certainty and false classi cations can produce large errors during classi cation because they are treated as negative examples. Smyth et al. in [48] discussed such issues, using the above problem as a case study. The paper's main contribution is the modeling and treatment of subjective label information given by the experts using probabilistic models. This research is important because it concludes that it is possible for the knowledge discovery methods to be modi ed to handle the lack of absolute ground truths.

In another study [14] - the Second Palomar Obser- vatory Sky Survey (POSS-II) - decision tree methods were also used for the classi cation of galaxies, stars and other stellar objects. About 3 TB of sky images were analyzed. Data images were preprocessed by low- level image processing system FOCAS, which selected objects and produced basic attributes like: magnitudes, areas, intensity, image moments, ellipticity, orientation, etc. Objects in the training data set were classi ed by astronomers. Based on this classi cation, about ten training sets for decision tree algorithm were con- structed. From the trees obtained by the learning al- gorithm, a minimal set of robust, general and correct rules was found. If no additional attributes describing features of a single image plate were used, the accuracy was about 75%. Additional attributes were de ned to reach a higher level of accuracy in every image. "Sure- stars" were detected in every image for the purpose of
nding image resolution. To gain e ciency, this process was also automated. Using "sure-stars", two additional attributes for every image plate were computed: resolu- tion scale and resolution fraction. These two attributes were used for normalization of attributes describing ob- jects produced by FOCAS. Other attributes like back- ground level or average intensity were also used to nor- malize plates. After the normalization the classi cation accuracy increased to about 94%. About 5 10^8 objects were classi ed. Obtained resolution was one magnitude better than the previous astronomical studies and it was possible to classify objects with images too faint to be classi ed by astronomers. The performance of decision tree methods was compared with neural networks. The tested neural networks algorithms were (a) traditional backpropagation, (b) conjugate gradient optimization, and (c) variable metric optimization. The last two al- gorithms use numerical optimization methods to com- pute network weights. A number of di erent networks was tested. The performance was fairly unstable with

accuracy varying from 30% to 95%. Additional draw- back of neural networks was the requirement to specify internal parameters such as the number of hidden layers or size. For future investigation, testing of unsupervised clustering techniques is planned.

The above studies showed the problems related to dif- ferences between images. The necessity of \normaliza-tion" of plates was shown to improve intra- and inter- plate classi cation.

Another example of image database mining is Stolorz et al's [49] study of fast spatio-temporal data mining from geophysical data sets. The authors describeda distributed parallel querying and analysis environ- ment called CONQUEST (CONtent-based QUErying in Space and Time). CONQUEST can be distinguished from other image database mining tools as it takes into account also temporal components of the datasets and it is designed to take advantage of parallel and distributed processing. CONQUEST was tested on two large cli- mate datasets[6] to detect cyclones and blocking features. The authors used heuristic rules based on signal process-ing methods for the extraction of characteristic weather phenomena. Di erent task decomposition methods were used to facilitate the distribution of work among a group of machines. In the case of cyclone detection, the opti- mal solution was the decomposition into separate tem- poral slices. The decomposition in the temporal dimen- sion is not always the best solution, especially when the state plays an important role in the detection of charac- teristic features. For detection of blocking features, the spatial decomposition, which assigns di erent blocks of grid points to di erent machines, was proven to be opti- mal. After detection of weather phenomena the authors used a clustering algorithm for the detection of shared spatial features. The goal of the authors is the building of a system that combines easy formulated queries with fast parallel execution and visualization of results for re nements of the queries.

2.6 Other Methods

The problem introduced by Fayyad et al.'s [14, 15] has been followed up by other researchers as well. One interesting study was done by Bell et al. [7] who proposed a method for knowledge discovery in spatial databases based upon evidence theory [21]. The authors took the image database mining problem described above as a case study. In this study [7] the authors described an extension of general framework for database mining in relational databases based on evidential theory [3] to mine knowledge from spatial databases.

Evidential reasoning [21] is a generalization of con- ventional probability in the sense that it does not make any assumptions about the independence of data be- ing analyzed. Therefore, the evidential reasoning may be a better choice than using probabilistic model like the Bayesian method to model the data like Venus pic- tures, where pixels may be interrelated. Evidential the- ory provides a method to combine evidences gathered from di erent sources to produce a single measure of un- certainty. Thus, it is claimed to be a better method to reason about spatial data in the presence of uncertainty. The combination of evidences is done using a technique based upon Dempster-Shafer theory. Informally, this theory can be regarded as a generalization of the con- ventional probability theory, where the probabilities are
 xed and known in advance, to the case where only the upper and lower bounds on probabilities are available [21]. Bell et al. [7] gave an example of how this method can be applied to image databases.

Major et al. [36] used IXL™ commercial tool for mining of the tropical storm database. The goal was to predict if hurricanes can reach the U.S. territory. Data describing hurricanes were decomposed to obser- vations at points. These observations were stored in a traditional relational database. Attributes like po- sition of the hurricane, speed, direction, angle to the coast, etc. were used. Since multiple tuples describ- ing the single hurricane in di erent points were stored, some data were interdependent. The interdependency of data causes problems, because the algorithm which was used assumes independence of data. The best rules according to di erent criteria like performance, novelty, signi cance and simplicity were selected from rules de- rived by the IXL. The GIS system was used to supportthe selection of the best rules. This study shows the necessity of extension of traditional data mining tech- niques toward spatial data mining for better analysis of complex spatial phenomena and spatial objects.

3 Future Directions

As we mentioned earlier, data mining is a young eld go- ing back no more than the late 1980s. Spatial data min- ing is even younger since data mining researchers rst concentrated on data mining in relational databases. Many spatial data mining methods we analyzed actually assume the presence of extended relational model for spatial databases. But it is widely believed that spatial data are not handled well by relational databases. As advanced database systems, like Object-Oriented (OO), deductive, and active databases are being developed, methods for spatial data mining should be studied in these paradigms.

Data mining in spatial object-oriented databases: How can the OO approach be used to design a spatial database [40, 42] and how can knowledge be mined from these databases? It is an important question since many researchers have pointed out that OO database may be a

[6] The datasets were chosen so that they were free of incomplete,noisy and contradictory data.

better choice for handling spatial data rather than tradi- tional relational or extended relational models. For ex- ample, rectangles, polygons, and more complex spatial objects can be modeled naturally in OO database. OO database techniques are maturing. OO knowledge rep- resentation techniques for spatial data have been pro- posed Mohan and Kashyap [40], and e cient SAM, like R-trees can be used to make OO database more e cient in access and retrieval of data. Therefore, exploiting OO technology in data mining is an area with enormous po- tential. Techniques for generalizations of complex data objects, methods and class hierarchies have been stud- ied by Han et al. [28].

Mining under uncertainty: Use of evidential rea- soning [21] can be explored in the mining process for image databases and other databases where uncertainty modeling has to be done. As mentioned in Bell et al's [7], evidential theory can model uncertainty better than traditional probabilistic models, like Bayesian methods. Fuzzy sets approach was applied to spatial reasoning [10, 11] and it can be extended to spatial data mining.

Alternative clustering techniques: Another in- teresting future direction is the clusterings of possibly overlapping objects like polygons as opposed to the clus-tering of points. Clusters can also maintain additional information about each object they contain, which can be the degree of membership. In this way, fuzzy clus- tering techniques can be used to accommodate objects having the same distance from the medoid.

Mining Spatial Data Deviation and Evolution Rules: One extension of current work in spatial data mining toward spatio-temporal databases is to study data deviation and evolution rules. For example, we can nd spatial characteristic evolution rules which summarizes the general characteristics of the changing data. During the mining process one can discover properties of the regions with average growth of crops over 2% per year. A spatial discriminant evolution rule discriminates the properties of objects in the target class from those in the contrasting classes. For example, one can make a comparison of the areas where air pollution increased last year with the areas where the air qualityhas been improved.

These rules may be used, for example, in medical imaging, where one would like to nd out how certain features are deviating from the norm or how they are evolving over time. Other applications may include, dis- covering and predicting weather patterns of geographic regions, land use planning, and others.

Using Multiple Thematic Maps: We discussed generalization-based methods which used a single the- matic map during generalizations. Various applica- tions demand spatial data mining to be conducted using multiple thematic maps. This would involve not only clustering but also spatial computations like map over- lay, spatial joins, etc. For example, to extract general weather patterns, it may be better to use temperature and precipitation thematic maps and to carry out gen-eralization in both.

Interleaved generalization: To extend the gene- ralization-based methods, it is interesting to consider in- terleaving spatial and nonspatial generalizations to get the results in more e cient manner. E cient process- ing can be achieved because usually spatial operations, like joins and overlays, are more expensive than non- spatial computations. Thus, by rst generalizing the non-spatial component and minimally using spatial gen- eralizations one may save a lot of computation time.

Generalization using temporal spatial data: This relates to the point we raised on discovery of data evolution rules earlier in this section. It may involve generalization over a sequence of maps collected during di erent time intervals. Then, comparison or summarization can be done to discover data evolutionregularities.

Parallel Data Mining: Due to the high volume of spatial data used during the computations mining using parallel machines or distributed farms of workstations can accelerate signi cantly the work. We expect that parallel knowledge discovery will be a growing research issue in both relational and spatial data mining.

Cooperation between Statistical Analysis and Data Mining: The enhancement of data mining tech- niques with mature statistical methods may produce interesting new techniques which may work well with di erent kinds of problems and on di erent data. For example, the statistical techniques may help in judge- ment on interestingness and signi cance of rules.

Spatial Data Mining Query Language: Design of the user interface can be one of the key issues in the popularization of knowledge discovery techniques. One can create a query language which may be usedby non-database specialists in their work. Such a query interface can be supported by Graphical User Interface (GUI) which can make the process of query creation much easier. Due to the special nature of data the query language can include features for display of the results of a query in graphical mode. The user interface can be extended by using pointing devices for the selection of objects of interest. The analysis of the results from the query may give feedback for re nement of the queries and show the direction of further investigation. The language should be powerful enough to cover the number of algorithms and large variety of data types stored in spatial databases.

Multidimensional rule visualization: Discovering knowledge is not enough because it has to be presented in a manner that the user can understand easily. One ofthe most e ective ways of digesting the rules discovered is through graphical visualizations. Humans are very good at interpreting visual data and scenes. This fact should be exploited in the data mining process. Multidimensional data visualization has been studied [32], but multidimensional rule visualization is still animmature area. Spatial data mining can use some well- developed visualization techniques in computer graphicsin this case.

4 Conclusion

We have shown that spatial data mining is a promising eld of research with wide applications in GIS, med- ical imaging, robot motion planning, etc. Although, the eld is quite young, a number of algorithms and techniques have been proposed to discover various kinds of knowledge from spatial data. We surveyed existing methods for spatial data mining and mentioned their strengths and weaknesses. This led us to future direc- tions and suggestions for the spatial data mining eld in general. The variety of yet unexplored topics and prob- lems makes knowledge discovery in spatial databases an attractive and challenging research eld. We believe that some of the suggestions that we mentioned have already been thought about by researchers and work may have already started on them. But what we hope to achieve is to give the reader a general perspective of the eld.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules Between Sets of Items in Large Databases. In Proc. 1993 ACM-SIGMOD Int. Conf. Management of Data, pp. 207{216, Washington, D.C., May 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In Proc. 1994 Int. Conf. VLDB, pp. 487{499, Santiago, Chile, Sept. 1994.

[3] S. S. Anand, D. A. Bell, and J. G. Hughes. A general framework for database mining based on evidential theory. Internal Report, Dept. of Inf. Sys., Univ. of Ulster at Jordanstown, 1993.

[4] W. G. Aref and H. Samet. Extending DBMS with Spatial Operations. In Proc. 2nd Symp. SSD'91, pp. 299{318, Zurich, Switzerland, Aug. 1991.

[5] W. G. Aref and H. Samet. Optimization Strategies for Spatial Query Processing. In Proc. 17th Int. Conf. VLDB, pp. 81{90, Barcelona, Spain, Sept. 1991.

[6] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The R-tree: An E cient and Robust Ac- cess Method for Point and Rectangles. In Proceedings of 1990 to ACM-SIGMOD Intl. Conf. on Management of Data, pp. 322-331, Atlantic City, USA, May 1990.*

[7] D. A. Bell, S. S. Anand, and C. M. Shapcott. Database Mining in Spatial Databases. International Workshop on Spatio-Temporal Databases, 1994.

[8] T. Brinkho , H.-P. Kriegel, and B. Seeger. E cient Processing of Spatial Joins Using R-trees. In Proc. 1993 ACM-SIGMOD Conf. Management of Data, pp. 237{ 246, Washington, D.C., May 1993.

[9] D. K. Y. Chiu, A. K. C. Wong, and B. Cheung. A Statistical Technique for Extracting Classi catory Knowledge from Databases. In Piatetsky-Shapiro and Frawley [43], pp. 125{141.

[10] S. Dutta. Qualitative Spatial Reasoning: A Semi-quantitative Approach Using Fuzzy Logic. In Proc. 1st Symp. SSD'89, pp. 345{364, Santa Barbara, CA, July 1989.

[11] S. Dutta. Topological Constraints: A Representational Framework for Approximate Spatial and Temporal Reasoning. In Proc. 2nd Symp. SSD'91, pp. 161{182, Zurich, Switzerland, August 1991.

[12] M. J. Egenhofer. Reasoning about Binary Topological Relations. In Proc. 2nd Symp. SSD'91, pp. 143{160, Zurich, Switzerland, August 1991.

[13] M. Ester, H.-P. Kriegel, and X. Xu. Knowledge Discov- ery in Large Spatial Databases: Focusing Techniques for E cient Class Identi cation. In Proc. 4th Int. Symp. on Large Spatial Databases (SSD'95), pp. 67{82, Port- land, Maine, August 1995.

[14] U. Fayyad, et al. Automated Analysis of a Large-Scale Sky Survey: The SKICAT System. In Proc. 1993 Knowledge Discovery in Databases Workshop, pp. 1-13, Washington, D.C., July 1993.

[15] U. M. Fayyad and P. Smyth. Image Database Ex- ploration: Progress and Challenges. In Proc. 1993 Knowledge Discovery in Databases Workshop, pp. 14- 27, Washington, D.C., July 1993.

[16] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, Menlo Park, CA, 1996.

[17] D. Fisher. Improving Inference through Conceptual Clustering. In Proc. 1987 AAAI Conf., pp. 461{465, Seattle, Washington, July 1987.

[18] D. Fisher. Optimization and Simpli cation of Hierar- chical Clusterings In Proc. 1st Int. Conf. on Knowl- edge Discovery and Data Mining (KDD'95), pp. 118{

123, Montreal, Canada, Aug. 1995.

[19] S. Fotheringham and P. Rogerson. Spatial Analysis and GIS, Taylor and Francis, 1994.

[20] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus. Knowledge Discovery in Databases: An Overview. In Piatetsky-Shapiro and Frawley [43], pp. 1{27.

[21] J. Guan and D. Bell. Evidence Theory and its Applications, vol. I. North-Holland, 1991.

[22] R. H. Guting. An introduction to spatial database systems. In VLDB Journal, 3(4):357{400, October 1994.

[23] R. Guttman. A dynamic index structure for spatial searching. In Proc. ACM SIGMOD Int. Conf. on Management of Data , Boston, MA, 1984, pp. 47-57.

[24] J. Han, and Y. Fu. Exploration of the Power of Attribute-Oriented Induction in Data Mining. In [16].

[25] J. Han, Y. Cai, and N. Cercone. Data-driven Discovery of Quantitative Rules in Relational Databases. IEEE Trans. Knowledge and Data Eng., 5:29{40, 1993.

[26] J. Han and Y. Fu. Dynamic Generation and Re ne- ment of Concept Hierarchies for Knowledge Discovery in Databases In Proc. AAAI'94 Workshop on Knowl- edge Discovery in Databases (KDD'94), pp. 157{168, Seattle, WA, July 1994.

[27] J. Han and Y. Fu. Discovery of Multiple-level Asso- ciation Rules from Large Databases. In Proc. 1995 Int. Conf. Very Large Data Bases, pp. 420{431, Zurich, Switzerland, September 1995.

[28] J. Han, S. Nishio, and H. Kawano. Knowledge Discovery in Object-Oriented and Active Databases. In F. Fuchi and T. Yokoi (eds), Knowledge Building and Knowledge Sharing, Ohmsha/IOS Press, pp. 221{230, 1994.

[29] M. Holsheimer and M. Kersten. Architectural Support for Data Mining. In CWI Technical Report CS-R9429, Amsterdam, The Netherlands, 1994.

[30] M. Holsheimer and A. Siebes. Data mining: The search for knowledge in databases. In CWI Technical Report CS-R9406, Amsterdam, The Netherlands, 1994.

[31] L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.

[32] D. Keim, H.-P. Kriegel, and T. Seidl. Supporting Data Mining of Large Databases by Visual Feedback Queries In Proc. 10th of Int. Conf. on Data Engineering, Houston, TX, pp. 302-313, Feb. 1994.

[33] E. Knorr and R. T. Ng. Applying Computational Geometry Concepts to Discovering Spatial Aggregate Proximity Relationships. In Technical Report, Univer- sity of British Columbia, 1995.

[34] K. Koperski and J. Han. Discovery of Spatial As- sociation Rules in Geographic Information Databases. In Proc. 4th Int'l Symp. on Large Spatial Databases (SSD'95), pp. 47{66, Portland, Maine, August 1995

[35] W. Lu, J. Han, and B. C. Ooi. Discovery of General Knowledge in Large Spatial Databases. In Proc. Far East Workshop on Geographic Information Systems pp. 275-289, Singapore, June 1993.

[36] J. Major, and J. Mangano. Selecting among Rules Induced from a Hurricane Database. In Proc of 1993 KDD Workshop, pp. 28{47, Washington, DC, July, 1993.

[37] C. J. Matheus, P. K. Chan, and G. Piatetsky-Shapiro. Systems for Knowledge Discovery in Databases. In IEEE Trans. Knowledge and Data Engineering, 5:903{913, 1993.

[38] R. S. Michalski, J. M. Carbonnel, and T. M. Mitchell, editors. Machine Learning: An Arti cial Intelligence Approach. Morgan Kaufmann, Los Altos, CA, 1983 .

[39] T. M. Mitchell. Generalization as Search. In Arti cial Intelligence, 18:203{226, 1982.

[40] L. Mohan and R. L. Kashyap. An Object-Oriented Knowledge Representation for Spatial Information. In IEEE Transactions on Software Engineering, 5:675{ 681, May 1988.

[41] R. Ng and J. Han. E cient and e ective clustering method for spatial data mining. In Proc. 1994 Int. Conf. Very Large Data Bases, pp. 144{155, Santiago, Chile, September 1994.

[42] P. van Oosterom and J. van den Bos. An Object-oriented Approach to the Design of Geographic Infor- mation System. In Proc. 1st Symp. SSD'89, pp. 255- 269, Santa Barbara, CA, July 1989.

[43] G. Piatetsky-Shapiro and W. J. Frawley, editors. Knowledge Discovery in Databases. AAAI/MIT Press, Menlo Park, CA, 1991.

[44] F. Preparata and M. Shamos. Computational Geometry: An Introduction. Springer-Verlag, New York, 19 85.

[45] J. R Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1992.

[46] H. Samet. The Design and Analysis of Spatial Data Structures. Addison-Wesley, 1990.

[47] G. Shaw, and D. Wheeler. Statistical Techniques in Geographical Analysis. London, David Fulton, 1994 .

[48] P. Smyth, M. C. Burl, U. M. Fayyad, and P. Perona. Knowledge Discovery in Large Image Databases: Deal- ing with Uncertainties in Ground Truth. In Proc. of AAAI-94 workshop on KDD, pp. 109{120, Seattle, WA, July 1994.

[49] P. Stolorz et al. Fast Spatio-Temporal Data Mining of Large Geophysical Datasets. In Proc. of the First International Conference on Data Mining KDD-95, pp. 300{305, Montreal, Canada, August 1995.

[50] M. Stonebraker. Readings in Database Systems. Morgan Kaufmann, 1988.

[51] M. Stonebraker. Readings in Database Systems, 2ed.. Morgan Kaufmann, 1993.

[52] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: an E cient Data Clustering Method for Very Large Databases. In Proc. 1996 ACM-SIGMOD Int. Conf. Management of Data, Montreal, Canada, June 1996. (to appear)