Kvisimine Applied To Problems In Geographical Information System

***Mrs. Sarita Patel**
****Prof. Deepa Chaurse**

Abstract

Images are highly complex multidimensional signals, with rich and complicated information content in geographical information system. For this reason they are difficult to analyze through a unique automated approach. However a KVISIMINE scheme is helpful for the understanding of image content and data content. In this paper, describes an application K-MEAN clustering algorithm and image information mining for exploration of image information and large volumes data. Geographical Information System, is any system that captures, stores, analyzes, manages, and presents data that are linked to location. Technically, a GIS is a system that includes mapping software and its application to remote sensing, land surveying, aerial photography, mathematics, photogram metric, geography, and tools that can be implemented with GIS software Building a GIS is a fruitful area if one likes the challenge of having difficult technical problems to solve. Some problems have been solved in other technologies such as CAD or database mana- gement. However, GIS throws up new demands, therefore requiring new solutions. This paper has examine difficult problems, and to be solved and gives some indication of the state of the art of current solutions.

Keywords: *K-MEAN Clustering, Visimine, image database, Image Information Mining, s-plus.*

1. Introduction:

The purpose of this paper is to present K-MEAN clustering algorithm is to discover internal structure in some set of data points - you supply the points and the number of clusters you expect to get, and the algorithm returns the same points, organized into clusters by proximity. And k-mean algorithm is a non-hierarchical approach to forming good clusters is to specify a desired number of clusters, say, k, then assign each case (object) to one of k clusters so as to minimize a measure of dispersion within the clusters. A very common measure is the sum of distances or sum of squared Euclidean distances from the mean of each cluster. The problem can be set up as an integer programming problem but because solving integer programs with a large number of variables is time consuming, clusters are often computed using a fast, heuristic method that generally produces good (but not necessarily optimal) solutions. The k-means algorithm is one such method. K-Means Training starts with a single cluster with its center as the mean of the data. This cluster is split into two and the means of the new clusters are iteratively trained. These two clusters are again split and the process continues until the specified number of clusters is obtained. If the specified number of clusters is not a power of two, then the nearest power of two above the number specified is chosen and then the least important clusters are removed and the remaining clusters are again iteratively trained to get the final clusters.When the user specifies random start the algorithm generates the k cluster centers randomly and goes ahead by fitting the data points in those clusters. This process is repeated for as many random starts as the user specifies and the Best value of start is found. The outputs based on this value are displayed.

The purpose of the algorithm is to discover internal structure in some set of data points - you supply the points and the number of clusters you expect to get, and the algorithm returns the same points, organized into clusters by proximity. Once you have the clusters, you can get their sample means, their variances; do a bunch of statistics, etc. This approach has become very

popular among the bioinformatics crowd, and especially among analysts of gene expression (micro array) data. And the VisiMine system provides the infrastructure and methodology required for the analysis of satellite images. In order to facilitate the analysis of large amounts of image data, we extract features of the images. Large images are partitioned into a number of smaller, more manageable image tiles. VisiMine uses an SQL-like query language that enables specification of the data mining task, features to be used in the mining process, and any additional constraints. VisiMine data can be accessed from within S-PLUS by using Java connectivity for images and ODBC connectivity for image and region data. In addition, VisiMine has the S-PLUS command tool, which provides for easy transfer of images to S-PLUS, and for data processing using the S-PLUS language. The S-PLUS images can be returned to VisiMine and displayed there. The rich statistical functionality of S-PLUS, together with the VisiMine user interface and the scalability of its data mining engine, allows for easy and powerful customization of the data analysis process.

This paper study the problem of GIS, GIS is a fruitful area if one likes the challenge of having difficult technical problems to solve. Some problems have been solved in other technologies such as CAD or database management. However, GIS throws up new demands, therefore requiring new solutions. This paper has chosen to examine difficult problems, to be solved and gives some indication of the state of the art of current solutions. The subject of Geographical Information Systems has moved a long way from the time when it was thought to be concerned only with digital mapping. Whereas digital mapping is limited to solving problems in cartography, GIS is much more concerned with the modeling, analysis and management of geographically related resources. At a time when the planet is coming under increasing pressure from an ever more demanding, increasing population, the arrival of GIS technology has come none too soon.

However, there is a widespread lack of awareness as to the true potential of GIS systems in the future. When the necessary education has been completed, will the systems be there to handle the challenge? It has to be said that the perfect GIS system has not yet been developed. The volume of data required representing the resources and infrastructure of a municipality is of the order of 0.5 to 1 gigabyte per 100,000 of population, supporting examples can be found where an urban region would require 14 million points simply to represent the map planimetry (Bordeaux, 1988). Not only are the municipal records required to be handled in the database, but also large amounts of map data in the form of coordinates, lines, polygons and associations of these. Today's database technology is barely up to the task of allowing the handling of geographic data by large numbers of users with adequate performance. Serious questions have been raised as to whether the most popular form of database, the relational model, will be able to handle the geometric data with adequate response. Certainly, if this data is accessed via the approved route of SQL calls, the achievable speed is orders of magnitude less than that which can be achieved by a model structure built for the task (Frank, 1988). It is a common problem with systems that contain parts that are front ended by different languages that it is not possible to integrate them properly. For example, a graphics system for mapping, which is "hooked into" a database, typically does not allow the full power of the database to be accessed from within the graphics command language, nor can the power of the graphics system be invoked from within the database query language. What is really needed is a system such that all data and functions can be accessed and manipulated in one seamless programming environment (Butler, 1988).Modern query languages such as SQL are not sufficient in either performance or sophistication for much of the major

development required in a GIS system - but then one would argue that they were not intended for this. One can see why people like SQL; it can give immense power in return for some fairly simple "select" constructs. A problem which has to be addressed is spatial queries within the language, since trying to achieve this with the standard set of predicates provided is extremely difficult and clumsy. (An example of a spatial query is to select objects "inside" a given polygon. If the route adopted is to provide two databases in parallel, a commercial one driven by SQL and a geometry database to hold the graphics, and then there is a problem constructing queries that address both databases. Ideally, the query language should be a natural subset of the front end language allowing access to the same seamless environment that the front end language provides. Much work needs to be done in the area of query languages for GIS.

Partitioning allows fetching of just the relevant tiles when retrieval of only part of the image is requested, and provides faster segmentation of image tiles. Individual image tiles are processed to extract the feature vectors. The VisiMine architecture distinguishes between pixel, region and tile levels of feature vectors. Pixel level features describe spectral and textural information about each individual pixel. Polygon level features describe connected groups of pixels. Following the segmentation process, each polygon is described by its boundary and by a number of attributes that present information about the content of the region in terms of shape, size, etc. The spectral and texture properties are based on pixel features of points within the polygon. Tile level features present spectrum and texture information about whole image tiles. All image features, together with the original images, are stored in a database system and indexed for fast retrieval. The auxiliary raster data such as Digital Elevation Models (DEM) can also be stored in the database and can be used for feature extraction and data analysis. The Oracle database system provides the means for fast information retrieval and network accessibility. Region level features can be stored in an ESRI Spatial Data Engine (SDE), and can then be displayed using ArcInfo or Arc View together with associated labels, features, or statistics. This storage functionality enables the fusion of GIS, optical, and DEM information for a variety of statistical analysis methods. The data mining power of VisiMine includes similarity searches on tile and polygon levels, clustering of tiles, classification and regression analysis, label training using Bayesian and tree models, and connection to S-PLUS with over 3000 statistical functions. The data mining queries are specified in an SQL-like language. A user may specify the features that are used in the mining task and constraints used to select data for the mining process. The graphical query constructor enables fast query creation by non-technical users. The user has high level of flexibility in choosing the features and images used for data analysis. The graphical user interface enables presentation of the models on high and general levels as well as drilling down into the details. The label training module enables interactive definition of models for land cover labels. The tile level summaries of pixel features are used for fast retrieval of tiles with high/low content of features and scenes with low confidence of classification. The initial model can be refined based on the feedback supplied by a data analyst who interactively trains the model using the system output and/or additional scenes. The experts may also refine models created by other users. The VisiMine system enables construction of sophisticated statistical models using the S- PLUS system, which can access data directly from the database, or using the GUI.

2. Related work: A great deal of research has been focused the **use of GIS in the spatial analysis of an archaeological cave site[1],** according to HOLLEY MOYES , archaeologists traditionally have viewed geographic information systems (GIS) as a tool for the investigation of large regions, its flexibility allows it to be used in non-traditional settings such as caves. This

study demonstrates the utility of GIS as a tool for data display, visualization, exploration, and generation. Clustering of artifacts was accomplished by combining GIS technology with a K-means clustering analysis, and basic GIS functions were used to evaluate distances of artifact clusters to morphological features of the cave. Results of these analyses provided new insights into ancient Maya ritual cave use that would have been difficult to achieve by standard methods of map preparation and examination.

The use of GIS in Archaeological Settlement Research Facts, Problems and Challenges [2], Frankfurt Germany, September 26th 2008 using Free and Open Source Software (FOSS) licenses generally allow free deployment anywhere and for any purpose. No *redundant* licensing costs, more flexible investment options, full control over development. Stable and long-lived data formats, free and *open standards* instead of "industry" no pressure to deprecate older software or data formats. No more *black boxes*, free knowledge and technology diffusion, no financial barriers to participation. Free and Open Source Software (FOSS) a paradigm and its relevance; debunking some myths FOSS GIS basic facts and history; GIS diversity; application vs. data centricity FOSS at Oxford Archaeology spatial data infrastructures (SDI)a free desktop GIS Applications in Landscape Archaeology (GRASS GIS)visibility, territories, spread, predictive models.. Reproducibility of research free knowledge and technology diffusion, no financial barriers to participation.

Clustering With GIS [3], Ece AKSOY, Turkey, presented there is no universally applicable clustering technique in discovering the variety of structures display in data sets. Also, a single algorithm or approach is not adequate to solve every clustering problem. There are many methods available, the criteria used differ and hence different classifications may be obtained for the same data. Grouping or classification of measurements is the key element in these data analysis procedures. There are lots of non-spatial clustering techniques in various areas. This study aims comparing different software in non-spatial and spatial clustering techniques, which can be used for different aims such as forming regional politics, constructing statistical integrity or analyzing distribution of funds, in GIS environment and putting forward the facilitative usage of GIS in regional and statistical studies. Self Organizing Maps (SOM) algorithm, which is the best and most common spatial clustering algorithm in recent years.

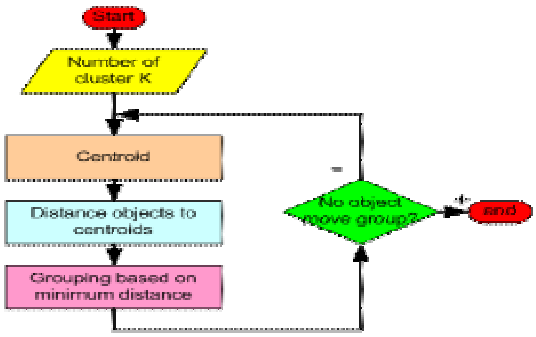
Geospatial Information and Geographic Information Systems (GIS): Current Issues and Future Challenges in June 8,2009[4],according to Peter Folger **,** Geospatial information is data referenced to a place a set of geographic coordinates which can often be gathered, manipulated, and displayed in real time. A Geographic Information System (GIS) is a computer system capable of capturing, storing, analyzing, and displaying geographically referenced information. Global Positioning System (GPS) data and their integration with digital maps has led to the popular handheld or dashboard navigation devices used daily by millions. For policy makers, this type of analysis can greatly assist in clarifying complex problems that may involve local, state, and federal government, and affect businesses, residential areas, and federal installations. Challenges to coordinating how geospatial data are acquired and used collecting duplicative data sets.

Implementation of the Extended Fuzzy C-Means Algorithm in Geographic Information Systems [5] Ferdinando Di Martino, Salvatore Sessa ,In 2009 , focused on density cluster methods have elevated computational complexity and are used in spatial analysis for the

determination of impact areas. We propose the extended fuzzy c-means (EFCM) algorithm like alternative method because it has three advantages: robustness to noise and outliers, linear computational complexity and automatic determination of the optimal number of clusters. We implement the EFCM algorithm inside geographic information systems (GIS) for the determination of buffer areas as hyper sphere volume prototypes which are circles in the case of bi-dimensional pattern data. In spatial analysis usually impact areas are determined by using density clustering algorithms which have an elevated computational complexity. Here we propose the EFCM algorithm because it has the following advantages: robustness to noise and outliers, linear computational complexity and automatic determination of the optimal number of clusters.

Issues of GIS data management [6], this paper deals with current issues of spatial data modeling and management used by spatial management applications. We use comparison of two main groups of software tools covering this area GIS and CAD systems - and the possibilities of their integration. Studying its functionality, we have found two main problematic issues. The first of them is the density distribution characteristics of stored data according to described area. CAD systems are oriented towards modeling individual man-made objects and structures with relatively high level of detail, so the data stored covers small areas with huge amount of information. Here the density distribution of data coverage is better balanced. So the combination of *described different densities is the first problem. The second* watched issue is the way of storing spatial data. While CAD data are usually stored in individual files (like DXF, IGES), GIS data tend to be stored in files or relational databases. The question we see is, if it is possible to store CAD data along with GIS data in the same database in spite of different distribution densities and different data models. paper describes ways of solving this problem. Now we can summarize the problem of the GIS and CAD integration. Because of the different characteristics of the GIS/CAD worlds, firstly there's need to decide for some suitable 3D data model, which could maintain complex and structured data types. This model also must be able to maintain the large-scale 3D models produced by CAD as well as low-scale objects used by GIS. An example of basic data model could be GeoToolKit. Secondly, there's need to prepare a system for maintaining the 3D data model. We suppose that this system has to be closely connected with an object-oriented.

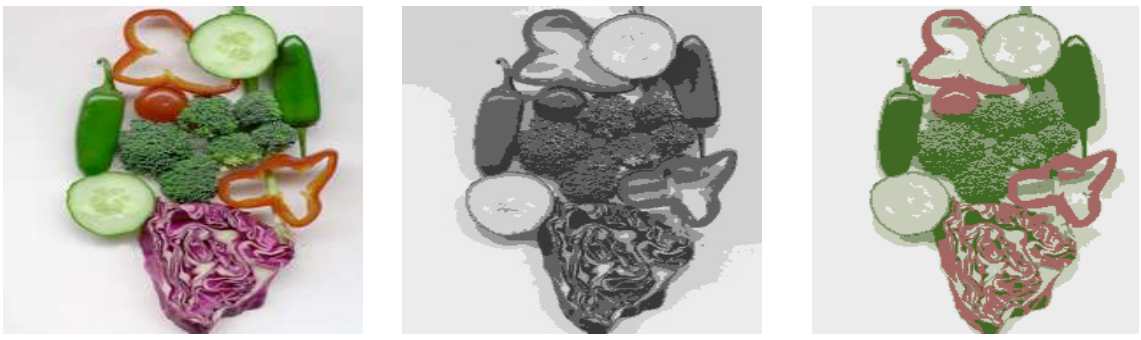
3. K-means clustering : K-MEANS clustering algorithms, the basic step of k-means clustering is simple. In the beginning we determine number of cluster K and we assume the centroid or center of these clusters. We can take any random objects as the initial centroids or the first K objects in sequence can also serve as the initial centroids. _Then the K means algorithm will do the three steps below until convergence Iterate until stable (= no object move group): Determine the centroid coordinate. Determine the distance of each object to the centroids Group the object based on minimum distance.



The numerical example below is given to understand this simple iteration. You may download the implementation of this numerical example as Matlab code here. Another example of interactive k-means clustering using Visual Basic (VB) is also available here. This shows the to use a database system for storage of the images and their features in order to overcome some limits on to the maximum size of files, and to benefit from indexing, query optimization, and partitioning features of the database. The image tiles and pixel level features are stored as BLOBs, with each band or feature stored in a separate column. The region and tile level features are stored in regular database tables that can be accessed easily for further processing.
In this algorithm:

- Choose k data points to act as cluster centers
- Until the clustering is satisfactory
  - Assign each data point to the cluster that has the nearest cluster center
  - Ensure each cluster has at least one data point splitting, etc
  - Replace the cluster centers with the means of the elements in the clusters

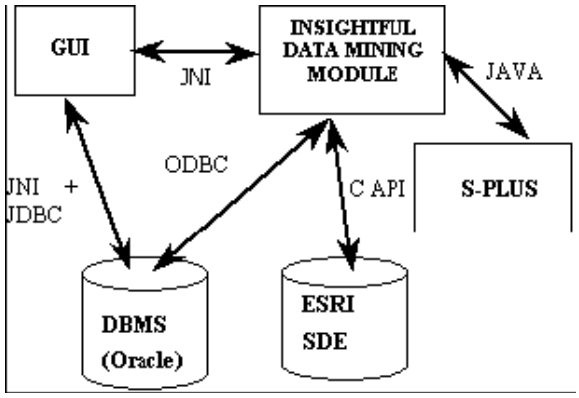
Image | Clusters on intensity | Clusters on color



K-means clustering using intensity alone and color alone

4. Database Organization:

VisiMine: VisiMine is a system for data mining and statistical analysis of large collections of remotely sensed images. VisiMine system provides the infrastructure and methodology required for the analysis of satellite images. Spatial information about region levels also can be stored in ESRI's Spatial Data Engine (SDE), together with the relevant GIS information. SDE provides open data access across local and wide area networks, and the Internet, using the TCP/IP protocol. This information can be combined with region level features such as texture, spectral properties, or DEM features. The SDE format allows a fusion of GIS, optical, and DEM information for a variety of visualization methods and data analysis functions. A mining process or a similarity search is initiated by submitting a query written in a data mining language similar to SQL. The query language allows the user to specify the type of knowledge to be discovered, the set of data relevant to the mining process, and the conditions that have to be satisfied by the data. Based on this query, an SQL statement is constructed to retrieve the relevant data. The data mining module processes the data and passes the information about the resulting tiles and regions to the GUI, which in turn directly retrieves the images from the database. The pabilities of the data mining engine are enriched through the Java connection to the S-PLUS statistical data analysis engine. The graphical user interface enables browsing and manipulation of the satellite images and associated features, creation of data mining queries, and visualization of the results of the data analysis.



The **VisiMine architecture** supports three levels of features: pixel, region,and tile level features. The feature extraction process starts with the analysis of spectral and textural properties at the pixel level. The numerical pixel data can be clustered in order to find a small number of classes. The VisiMine system supports extraction of the following features:

- Texture features using Gabor wavelets
- Clustering (spectral, textural, and others)
- Spectral Mixture Analysis features.
- Segmentation and shape descriptors of the regions.
- Spatial relationships between regions.
- Histograms, max, min, mean, and standard deviation of pixel features for each region and tile.

VisiMine uses an SQL-like query language that enables specification of the data mining task, features to be used in the mining process, and any additional constraints. The system is capable of performing similarity searches based on any combination of features.VisiMine allows weighting of the features. In addition to the similarity search, the VisiMine system provides functionality for other types of analyses of remotely sensed data. This functionality includes data

clustering, building regression and classification models, prediction of land cover types, summarizing data, searching using visual grammar and interactive label training using Bayesian and tree models. Interactive label training methods enable searches for features that are very difficult to describe analytically. In VisiMine we use a method for training of land cover labels that employs naïve Bayesian classifiers. VisiMine is based on decision tree models. V S-PLUS connectivity Insightful S-PLUS is an interactive computing environment for graphics, data analysis, statistics, and mathematical computing. VisiMine data can be accessed from within S-PLUS by using Java connectivity for images and ODBC connectivity for image and region data. In addition, VisiMine has the S-PLUS command tool, which provides for easy transfer of images to S-PLUS, and for data processing using the S-PLUS language. The S-PLUS images can be returned to VisiMine and displayed there. The VisiMine can also display S-PLUS graphics, which are created using a command line interface and shown within S-PLUS plot window. The combination of S-PLUS and VisiMine features creates a unique environment for interactive exploration and analysis of remotely sensed data. The rich statistical functionality of S-PLUS, together with the VisiMine user interface and the scalability of its data mining engine, allows for easy and powerful customization of the data analysis process.