Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 6 (2012) 452 – 459

Procedia Technology

2nd International Conference on Communication, Computing & Security [ICCCS-2012]

A Novel Spatial Fuzzy Clustering using Delaunay Triangulation for Large Scale GIS Data (NSFCDT)

Parthajit Roy[a], J.K. Mandal[b*]

[a]The Dept. of Comp. Sc., The University of Burdwan, Burdwan, West Bengal, India-713104
[b]The Dept. of Comp. Sc. & Engg., The University of Kalyani, West Bengal, India-741235

Abstract

This paper proposed a novel fuzzy clustering method on spatial data based on Delaunay Triangulation. Given a set of points in a two dimensional space, the objective is to group the data into several sets. The model uses two passes. In the first pass, the underlying model used a Delaunay Triangulation method for initial spatial clustering. The clustering, in this pass, is done on the notion of proximity of points. The Fuzzy method is applied as a post processing refinement of the clusters in the second pass. The underlying mathematical backbone of the proposed model along with the correctness of the algorithm has also been conformed.

Keywords: Spatial Clustering; Fuzzy Refinement; Delaunay Triangulation; Two Pass Clustering; NSFCDT.

1. Introduction

Pattern classification (?) is one of the major emerging fields in today's computation. Pattern classification using spatial clustering is very popular and widely applicable to a variety of areas of computer science and computer applications. Given a set of points in a space, the objective of clustering is to group the data points into different sets depending upon various underlying attributes and properties. Density based clustering (?) is very fundamental and the base of the other clustering techniques.

Geographical Information System (GIS) is another emerging field (?) (?). Clustering in GIS is even more challenging because GIS deals with huge data. So, the clustering as well as data structure for storing the knowledge about cluster is one of the major concerns of literally every GIS system.

This paper proposed a novel two pass fuzzy clustering method based on Delaunay Triangulation (DT) and applied it on GIS data. The Delaunay Triangulation (?) is a planner decomposition of spatial data. Given a set of data points, the DT decomposes the set into triangles based on distance vector. The DT has several important properties out of which the proposed model of the paper used mainly the notion of proximity of Delaunay Triangles. Some of the

[*] Corresponding author. Tel.: +91-33-2580-9617 ; fax: +91-33-2580-9617.
E-mail address: jkm.cse@gmail.com

doi:10.1016/j.protcy.2012.10.054

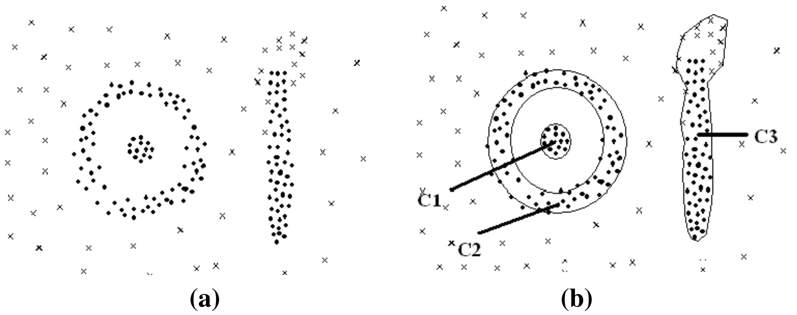

Fig. 1. (a) Set of Data Points along with noises. (b) Clustering based on Rational Measure.

material on Delaunay Triangulation and other Computational Geometry models are made online by Chew (**?**) and Gold (**?**). Some interesting applications of Delaunay-Voronoi model are also available (**?**).

The idea of clustering using Delaunay Triangulation is also proposed by Yang et al in their algorithm NSCABDT (**?**), where the classification parameters are set based on statistical measures. Present paper proposed a novel two pass model. The first pass is crisp and classifies the spatial points on the proximity property of Delaunay Triangulation.

In the second pass of the computation, the model used fuzzy logic(**?**) to perform a final refinement on the clustering and incorporates some extra points depending upon the nature of the cluster.

Section **??** deals with the first pass of the clustering algorithm, based on Delaunay Triangulation, along with the proof of its correctness. Section **??** described the second pass i.e. the fuzzy refinement and its underlying mathematics. This section also proposed the second pass algorithm for Fuzzy clustering. Section **??** discussed the result and outputs of the proposed model. Conclusion and the future scope is given in the section **??** and the references are drawn at the end.

2. Clustering Using Delaunay Triangulation

The clustering technique of the proposed model assumed two dimensional data points. Let there be n points $P = \{p_1, p_2, p_3, \ldots, p_n\}$ in a two dimensional space. In our case n is huge (Order of thousands) as we are dealing with GIS data. Moreover there are some noises distributed throughout the Area of Interest (AOI). The problem is to cluster the points into proper number of sets excluding the noise points. The nature of noise points and the nature of cluster points are same, i.e. for example, both cluster points and noise points may be Pine tree. The noise points are so named because they are sparsely distributed over the AOI and therefore may be stated that there existence in that place is not because of the property of the cluster but it is only due to an exception.

Our objective is to classify the point set in a rational way i.e., though from the Figure **??** (a) it is clear what will be clustering in the right hand side bar, the proposed model may include upper side noises also because from the picture it is clear that it may be a part of the cluster from rational point of view [Figure**??** (b)]. This final clustering will be done in the second pass.

In the first pass, the model classifies all of the cluster points using Delaunay Triangulation. The Delaunay Triangulation is a special kind of planner decomposition of a set of points of a two dimensional space where it will triangulate the entire area based on the given data points.

2.1. Delaunay Triangles and Some Important Concepts

Given a set of points $P = \{p_1, p_2, p_3, \ldots, p_n\}$ in a two dimensional space. The Delaunay Triangulation of the set is a triangulation of the point set having the following properties.

- Let $p_i, p_j, p_k \in P$ be any three points of the point-set, these will be the vertices of a Delaunay Triangle if and only if the circumcircle of the triangle formed by p_i, p_j, p_k containing no other point of the set P. i.e., the circumcircle of a Delaunay Triangle is empty.

- If two sites p_i and p_j are connected by an edge in of a triangle in Delaunay triangulation, they will be called proximity points.
- The total number of edges formed by Delaunay Triangulation is at most $3n-6$. As an edge can contribute to maximum two triangles, total number of triangles is of linear order of n . So, it can always be assured that this triangulation will not form too many triangles so that the performance of the model is dropped.
- The Delaunay Triangles are formed in such a way that it maximizes the minimum angles of the triangles over the DT set. So proximity will always be ensured.

Algorithm **??** is the proposed crisp clustering technique used as the first pass. It accepts a data set of n points and clusters them based on Delaunay Triangulation. The Delaunay Triangulation in the present model is done by Bowyer-Watson algorithm formulated by Bower (**?**) and Watson (**?**).

The Figure **??**(a), **??**(b), **??**(a) and **??**(b) shows a small input set and the intermediate as well as the final output generated through the algorithm. In Figure **??**(a) we have taken a few data points. Some of which are sparsely distributed as noise and few are densely placed. Figure **??**(b) shows a triangular decomposition of the given point sets. After the triangulation is complete, it starts clustering. For this, a benchmark size of the triangle, called the satisfiable size, is determined. All triangles with size greater than satisfiable size are discarded. Remaining triangles are considered for clustering. The model peaks up an arbitrary triangle from the satisfiable triangles at random and builds the cluster based on it and so on. Figure **??**(b) shows clustering on excluding the odd triangles.

Algorithm 1 Procedure Delaunay Classification

```
Input: $PointSet = \{p_1, p_2, \ldots, p_n\}$
    Declare DT As DelaunayTriangleSet
    $DT \leftarrow \textsc{DelaunayTriangles}(PointSet)$
    Declare ClusterSet As Set
    Declare NewSet As Set
    Declare TmpSet As Set
    for all dt In DT do
        if $\textsc{Visited}(dt) = False$ then
            $\textsc{Initialize}(NewSet)$
            $\textsc{Push}(dt, Stack)$
            while $\textsc{Empty}(Stack) = False$ do
                $Triangle\ T \leftarrow \textsc{Pop}(Stack)$
                if $\textsc{Satisfiable}(T)\&\textsc{NotVisited}(T)$ then
                    $T.Visited \leftarrow True$
                    $NewSet \leftarrow NewSet \cup \{T\}$
                    $TmpSet \leftarrow \textsc{Neighbors}(T)$
                    for all $Element$ In $TmpSet$ do
                        if $\textsc{NotVisited}(Element)$ then
                            $\textsc{Push}(Element, Stack)$
                        end if
                    end for
                end if
            end while
            $ClusterSet \leftarrow ClusterSet \cup NewSet$
        end if
    end for
```

2.2. *Correctness of Algorithm*

To prove the effectiveness of the algorithm, we shall first introduce some of the definitions. These are:

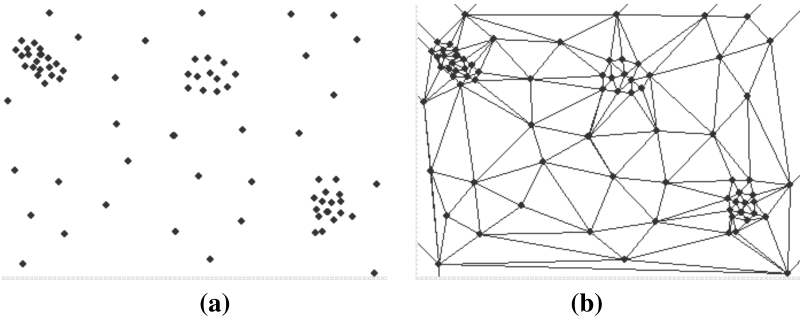

Fig. 2. (a) Set of Data Points to be clustered.(b) Delaunay Triangulation by the Algorithm **??**

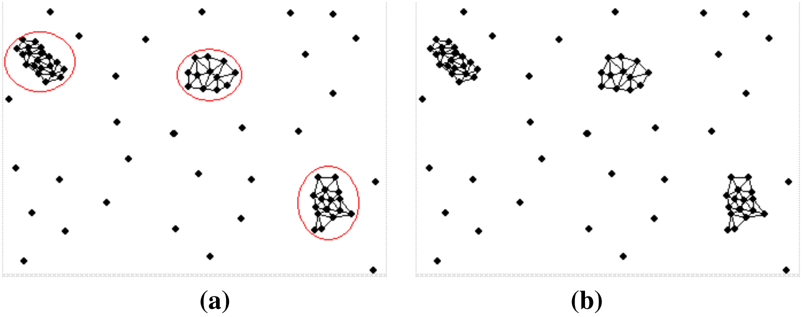

Fig. 3. (a) The output of the algorithm is highlighted. (b) The output of the algorithm produces three clusters.

Dirty Triangle and Satisfiable Triangle: A triangle t of Delaunay Triangle set T is said to be Satisfiable Triangle if the property P holds good for t and is denoted by Satisfiable $w.r.t.\ P$. Otherwise the triangle t will be called a dirty triangle and will be denoted by Dirty $w.r.t.\ P$.

Neighbor: For a given triangle t_1, a triangle t_2 will be called a neighbor of t_1, if t_1 and t_2 shares a common edge. Trivially if t_1 and t_2 are equal, i.e. t_1 will be called a neighbor of itself as it shares its edges with itself.

P-Similar Neighbor: For a given property P and a given triangle t_1 in the Delaunay Triangle set T, a triangle t_2 will be called $P-Similar$ Neighbor of t_1 in T $w.r.t.$ property P if,

- t_1 is Satisfiable $w.r.t.\ P$.
- $t_2 \in T$
- t_2 is Satisfiable $w.r.t.\ P$
- t_2 is either a neighbor of t_1 or a neighbor of any $P-Similar$ neighbor of t_1.

P-Cluster w.r.t a triangle: for a given triangle t belonging to the DT set T, $P-Cluster$ of t, denoted by $P-Clast(t)$ can be defined as

- $P-Clast(t)$ is a non-empty set of triangles.
- t satisfies property P.
- $t \in P-Clast(t)$
- For any $s \in T$, $s \in P-Clast(t)$ if and only if s is $P-similar$ neighbor of t in T.

2.2.1. The Notion of Correctness

Algorithm **??** will be called a correct algorithm $w.r.t.$ the property Satisfiable Triangle (as property P), if it produces a set of m Clusters $C_1, C_2, C_3, \ldots, C_m$ of Triangles satisfying the following properties,

- $C_i \subseteq T; \forall i = 1, 2, 3, \ldots m$

- $\bigcup_{i=1,2,3,\ldots,m} C_i = AllSatisfiableTriangles$
- $C_i \bigcap C_j = \phi; \forall i \neq j$
- $\forall DirtyTriangles\ d, d \notin \bigcup_{i=1,2,3,\ldots,m} C_i$
- For any two arbitrary triangles t_i, t_j; t_i, t_j belongs to the same cluster C; for some C in $C_1, C_2, C_3, \ldots, C_m$, $iff\ t_i$ is $P-Similar$ Neighbor of t_j or vice-versa.

Corollary 1 *If we start with any arbitrary triangle of a Cluster, we will always end up with the same cluster set.*

Proof: To prove this, we shall take the help of equivalence relation.

Let R be relation defined as follows: $t_1Rt_2 \Rightarrow t_1, t_2 \in DT$ and t_1, t_2 are $P-Similar$ neighbor of each other. We shall prove that R is an equivalence relation.

Reflexive: From the definition of $P-Similar$ neighbor, it is clear that $\forall t_1 \in DT \& P-Similar(t_1) = True \Rightarrow t_1Rt_2$.

Symmetric: If t_1 is $P-Similar$ to t_2 then from the definition it is clear that the converse is also true. i.e.$t_1Rt_2 \Rightarrow t_2Rt_1$

Transitive: Let there be three triangles t_1, t_2 and t_3 $s.t.$ t_1Rt_2 and t_2Rt_3 holds good. From relation R, it is clear that t_1 and t_2 are $P-Similat$ and so also t_2 and t_3. Now, from the definition of $P-Similar$ity it can be noted that, we may reach from t_1 to t_2 by satisfiable triangle searching only. Similarly we can reach from t_2 to t_3 by satisfiable triangle searching only. So, we can reach from t_1 to t_3 by searching satisfiable triangle only. So, t_3 is a $P-Similar$ neighbor of t_1 and hence there always exists a relation t_1Rt_3

So, R is an equivalence relation. Now, equivalence relation creates a partition and each partition is transitively close. So, if we take any arbitrary point and examine all of its neighbors and so forth, we will always end up with the same cluster and nothing extra or less.

The Corollary **??** solves the problem of arbitrary triangle selection in a random way. i.e. if we start with an arbitrary triangle of the cluster the cluster will always be unique.

Theorem 1 *The Algorithm produces correct number of clusters with correct cardinality of each of them.*

Proof: The satisfiable triangles form an equivalence relation and by Corollary **??**, it is clear that given an element of a cluster, it is possible to reach all other points of the same cluster by transitivity operation. The equivalence relation also guarantees that two partitions are disjoint. i.e. by any of reflexive, symmetric or transitive operations, it is not possible to reach from one partition to the other. The algorithm checks all of the neighbors of a given triangle and so on in a recursive transitive manner. So, it clearly identifies every member of a given cluster. Hence the right cardinality of the clusters is conformed. Lastly, as the algorithm checks every triangle of the DT set, it identifies all of the clusters correctly.

3. The Fuzzy Refinement

Fuzzy logic is applied in the second pass of the model to refine the clustering. The idea is that whenever a large cluster exists, the scattered data just outside the cluster may be treated as a part of the cluster. For example if there is a large forest of Pine tree, then the Pine trees just outside the forest may also be a part of the forest. So, instead the core dense part of the forest, the dense part as well as the shallow outer region together will form the forest. This realistic inclusion of the noises for rational classification is introduced in the model by the incorporation of Fuzzy logic. Some density based fuzzy pattern classification technique can be found in (**?**).

In the present paper we have considered the area of the cluster as a parameter. As the Delaunay Triangulation is a planner decomposition of the point set, the area of all of its triangles will be the area of the cluster. The idea is that, the larger the cluster area, the larger the shallow area which has to be incorporated in the cluster.

We define the fuzzy membership function as follows:

$$\mu(x:\alpha,\beta) = \begin{cases} 0 & , \quad x < \alpha; \\ 2\left(\frac{x-\alpha}{\beta-\alpha}\right)^2 & , \quad \alpha \leq x < \frac{\alpha+\beta}{2}; \\ 1-2\left(\frac{x-\beta}{\beta-\alpha}\right)^2 & , \quad \frac{\alpha+\beta}{2} \leq x < \beta; \\ 1 & , \quad x \geq \beta. \end{cases}$$

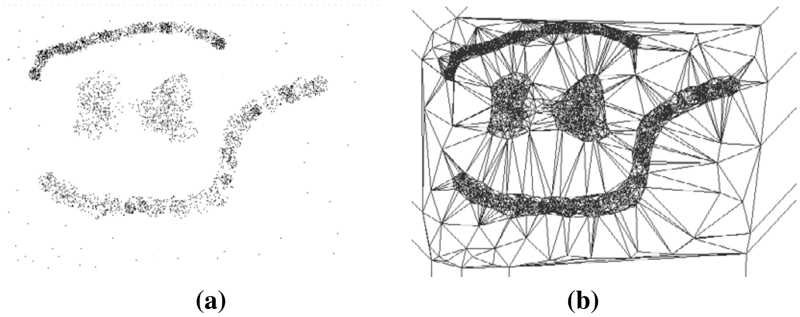

Fig. 4. (a) A Set of 4088 data points in a spatial domain. (b) The Delaunay Triangulation of the set of 4088 data points.

The μ in the paper is called the ***strength of inclusion***. The greater the value of μ the larger the area in the boundary the cluster will consider. The fuzzy parameters, α and β in this case, can be set according to the need of the particular application. As the membership value lies between 0 and 1, the degree can be multiplied by a suitable constant to make it rational. The Fuzzy based technique NSFCDT is illustrated in Algorithm **??**.

Algorithm 2 Procedure Fuzzy Based NSFCDT

```
Input: $PointSet = \{p_1, p_2, \dots, p_n\}$
BUILD( ClusterSet)                                   ▷ By Algorithm ??
for all $C_i$ In $ClusterSet$ do
    $Area \leftarrow$ GETAREA($C_i$)
    $Degree \leftarrow$ GETFUZZYDEGREE($Area$)
    SETSATISFIABILITYCONDITION($Degree$)
    for all $Triangle$ In $Cluster$ do
        $N \leftarrow Neighbour(Triangle)$
        for all $E$ In $N$ do
            if FUZZYSATISFIABLE($E$) then
                if $E \in C_j, \forall j = 1 \dots m \;\&\; j \neq i$ then
                    $C_i \leftarrow C_i \cup C_j$
                else
                    $C_i \leftarrow C_i \cup \{E\}$
                end if
            end if
        end for
    end for
end for
```

4. Results and Discussion

We have adapted Bowyer-Watson algorithm (**?**)(**?**) for Delaunay Triangulation. The crisp output as well as the fuzzy output shows very good results. We have compared NSFCDT with the NSCABDT (**?**). The Scheme has been tested with a series of data sets consisting of 5000, 6000, 7000, 8000, 9000 and 10000 points. In each and every case, the proposed NSFCDT shows reasonably good results. NSCABDT shows better performance when the number of clusters is less (less then 5). A Graphical representation is shown in Figure **??**(a),**??**(b), **??**(a) and **??**(b). In Figure **??**(a), a data set of 4088 points has been taken. Figure **??**(b) shows the Delaunay Triangulation of the data set. In Figure **??**(a), the Graphical result of crisp clustering is shown and in the Figure **??**(b) the final fuzzy output of the second pass of the model is shown. Figure **??**(b) shows, that though crisply there are four clusters, the fuzzy model produces three instead of four clusters as it seems from the Figure **??**(a).

The test results for various data sets are presented in the table **??**. The table clearly shows that the NSFCDT is

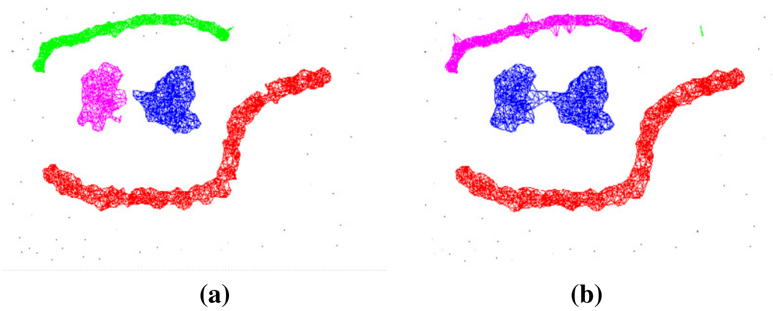

Fig. 5. (a) Crisp Clustering produces four different clusters (b) Fuzzy Clustering produces three different clusters.

superior in crisp clustering. As we have proposed a second pass for fuzzy refinement, there is slight overhead and the time complexity is slightly high. The interesting point is that, the little time overhead of the present model offering a fuzzy rationality which models the real life clustering better.

Table 1. A Comparative Study of NSCABDT and NSFCDT

No of Points	Run Time in Seconds		
	NSCABDT	NSFCDT	
		Crisp Clustering	Fuzzy Clustering
	Run Time	Run Time	Run Time
5000	48.2	40.04	55.18
6000	71.4	54.72	81.99
7000	93.8	63.19	87.45
8000	117.9	97.37	135.82
9000	151.3	129.48	183.88
10000	189.4	167.73	228.31

The proposed model has been applied on GIS data taken from Remote Sensing image of LANDSAT-7 (*Courtesy of Bidhan Chandra Krishi Viswavidyalaya, West Bengal, India*). A very satisfactory results are found on application of the techniques. A part of Sundarban area of West Bengal having number of rivers and a very dense forest, the model shows a very good result on a gray scale Remote Image. The Crisp classification shows several tree set area (Six large clusters). Even a very low aperture canal separates two clusters distinctly (The Green color and Red color clusters in fig **??**(a)). For Blue and Cyan cluster of fig **??**(a), a small area having shallow distribution separates the clusters. Both the cases are clearly not realistic (clear from fig **??**). The fuzzy inclusion gives a rational classification of the tree set areas and it is established from the algorithm that there are broadly four clusters instead of six, suggested by the crisp clustering. Figure **??**, **??**(a) and **??**(b) shows the results.

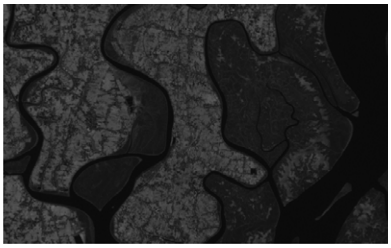

Fig. 6. A portion of Sundarban area of West Bengal, India.(*Courtesy of BCKV, West Bengal, India.*)

The notion of *density* is not same for dense forest and dense buildings in a city. Similarly inclusion factor will also change from application to application. Hence, the parameters like *proximity* and fuzzy *strength of inclusion* are

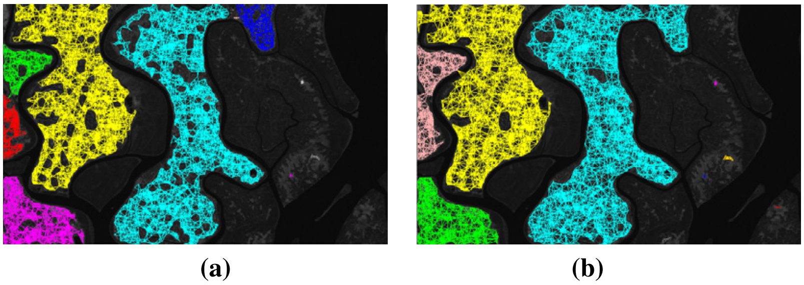

Fig. 7. (a) Crisp Clustering produces six main clusters. But clusters in Blue and Cyan are essentially same. The same is true for Green and Red Clusters also. (b) Fuzzy Clustering produces four different clusters. The Cyan and Blue are merged to form a single cluster and so also Red and Green and they have formd a single cluster shown in Pink.

subjective and have to be adjusted for a particular application.

5. Conclusion and Future Scope

The clustering using NSFCDT is highly efficient and it produces results with a very high degree of accuracy. The Bowyer-Watson algorithm obtains $O(n\sqrt{n})$ time complexity. Both first and second passes of proposed algorithm take extra $O(n)$ time. So the overall time complexity of the algorithm NSFCDT is $O(n\sqrt{n}+k_1n+k_2n) = O(n\sqrt{n})$, where k_1 and k_2 are constant factors of computation.

Though the performance of NSFCDT is highly satisfactory, nevertheless there is enough scope of betterment. Firstly the model cannot treat two triangles as neighbors if they share common vertex instead of common edge. This can be managed by running an extra pass. The use of even more efficient data structures can improve the performance of the model. NSFCDT is a generic clustering model. Customizing the model by using modern soft computing tools can make even better clustering in constrained specific cases. Some statistical model can also be used for performance tuning.

Acknowlwdgements

The authors express the deep gratitude to the Department of Computer Science, the University of Burdwan, Department of Comp. Sc. & Engg., Kalyani University and PURSE Scheme of DST, Government of India for providing necessary infrastructure and support for the work.

References

Bowyer, A. 1981. “Computing Dirichlet tessalations.” *The Computer Journal* 24(2):162–166.

Chew, L. Paul. N.d. “http://www.cs.cornell.edu/home/chew/chew.html; Access Date:24-04-2011.”.

de Berg, Mark, Otfried Cheong, Marc van Kreveld & Mark Overmars. 2008. *Computational Geometry-Algorithms and Applications*. 3 ed. Springer.

Duda, R.O., P.E. Hart & D.G. Stork. 2008. *Pattern Classification*. 2 ed. John Wiley.

Ester, M., H. P. Kriegel, J. Sander & X. Xu. 1996. “Density- based algorithm for discovering clusters in large spatial databases with noise,” Proceedings of the 1996 Knowl-edge Discovery and Data Mining (KDD’96)).” *International Conference,AAAI Press* pp. 226–231.

Gold, Chris. N.d. “http://www.voronoi.com/; Access Date:27-03-2011.”.

Gold, Christopher M. 1991. “Problems with Handling Spatial Data-The Voronoi Approach.” *CISM Journal ACSGC* 45(1):65–80.

Lillesand, T.M., R.W. Kiefer & J.W. Chipman. 2004. *Remote Sensing and Image Interpretation*. 5 ed. Wiley India.

Lo, C.P. & Albert K.W. Yeung. 2007. *Concepts and Techniques of Geographic Information Systems*. 2 ed. Prentice Hall of Iindia.

Rhee, Frank Chung-Hoon, Jong Hoon Park & Byung In Choi. 2007. Density Based Fuzzy Support Vector Machines for Multicategory Pattern Classification. In *Anal. and Des. of Intel. Sys. using SC Tech.* Vol. ASC(41) Springer-Verlag pp. 109–118.

Watson, D.F. 1981. “Computing The n-Dimensional Delaunay Tessalation with Application to Voronoi polytops.” *The Computer Journal* 24(2):167–172.

Yang, Xiankun & Weihong Cui. 2010. “A Novel Spatial Clustering Algorithm Based on Delaunay Triangulation.” *Journal of Software Engineering and Applications* 3:141–149.

Yen, John & Reza Langari. 1999. *Fuzzy Logic-Intelligence, Control, and Information*. Pearson Education.