A New and Efficient K-Medoid Algorithm for Spatial Clustering

Qiaoping Zhang and Isabelle Couloigner

Department of Geomatics Engineering, University of Calgary
2500 University Drive N.W.
Calgary, Alberta Canada T2N 1N4
{qzhang, couloigner}@geomatics.ucalgary.ca

Abstract. A new *k*-medoids algorithm is presented for spatial clustering in large applications. The new algorithm utilizes the TIN of medoids to facilitate local computation when searching for the optimal medoids. It is more efficient than most existing *k*-medoids methods while retaining the exact the same clustering quality of the basic *k*-medoids algorithm. The application of the new algorithm to road network extraction from classified imagery is also discussed and the preliminary results are encouraging.

1 Introduction

Clustering is the process of grouping a set of objects into classes or clusters so that objects within a cluster have similarity in comparison to one another, but are dissimilar to objects in other clusters [Han *et al* 2001]. In simpler words, clustering is a process of finding natural groupings in a set of data. It has been widely used in the applications such as marketing, city planning, insurance, medicine, chemistry, remote sensing, and so on. A complete review of the current state-of-the-art of spatial clustering can be found in [Han *et al*, 2001] or more recently, in [Guo *et al*, 2003].

In the field of automatic object extraction from remotely-sensed imagery, Doucette *et al* (1999; 2001) provided a self-organizing road map (SORM) approach to road centerline delineation from classified high-resolution Multi-Spectral Imagery (MSI). The SORM is essentially a spatial clustering technique adapted to identify and link elongated regions. This technique is independent from a conventional edge definition, and can meaningfully exploit multispectral imagery. Therefore, it has some promising advantages over other existing methodologies.

Spatial clustering techniques enable the identification and link of elongated road regions. Unfortunately, traditional *k*-means, Kohonen learning approaches, which were used by Doucette *et al* (1999, 2001), are sensitive to the noises in classified images. The *K-Medoids* approach is more robust in this aspect, but it is very time-consuming. We have to investigate more efficient spatial clustering algorithms in order to apply these techniques to a large image. This is the main motivation of this paper.

The remaining parts of this paper are organized as follows: The existing *k*-medoids methods are briefly introduced in the next section. The new efficient *k*-medoids algorithm is presented afterwards with some experiment results. The application of

O. Gervasi et al. (Eds.): ICCSA 2005, LNCS 3482, pp. 181 – 189, 2005.

the proposed algorithm in road network extraction is discussed and finally some conclusions are given.

2 *K*-Medoids Clustering Methods

The k-medoid method is one of the partitioning methods. Because it uses the most centrally located object (*medoids*) in a cluster to be the cluster centre instead of taking the mean value of the objects in a cluster, it is less sensitive to noise and outliners compared with the *k*-means approach [Han *et al*, 2001]. Therefore, the *k*-medoids method should be more suitable for spatial clustering purpose than the *k*-means method because of the better clustering quality it can achieve. However, it is well known that a *k*-medoids method is very time-consuming. This motivates us to investigate the possibility to improve the efficiency of the *k*-medoids method in the context of road network extraction.

2.1 PAM Algorithm

An early *k*-medoids algorithm called Partitioning Around Medoids (PAM) was proposed by Kaufman and Rousseeuw (1990). The PAM algorithm can be described as follow [Ng and Han, 1994]:

1. Select k representative objects arbitrarily.
2. Compute total cost TC_{ih} for all pairs of objects O_i, O_h where O_i is currently selected, and O_h is not.
3. Select the pair O_i, O_h which corresponds to $min_{\,Oi,\,Oh}\,(TC_{ih})$. If the minimum TC_{ih} is negative, replace O_i with O_h, and go back to Step (2).
4. Otherwise, for each non-selected object, find the most similar representative object. Halt.

Because of its complexity, PAM works effectively for small data sets (e.g., 100 objects in 5 clusters), but is not that efficient for large data sets [Han *et al*, 2001]. This is not too surprising if we perform a complexity analysis on PAM [Ng and Han, 1994]. In Step 2 and 3, there are altogether k $(n\text{-}k)$ pairs of $[O_i, O_h]$, where k is the number of clusters and n is the total number of objects. For each pair, computing TC_{ih} requires the examination of $(n\text{-}k)$ non-selected objects. Thus, Step 2 and 3 combined is of $O(k(n\text{-}k)^2)$. This is the complexity of only one iteration. Thus, it is obvious that PAM becomes too costly for large values of n and k [Ng and Han, 1994].

2.2 CLARA Algorithm

To deal with larger data sets, a sampling-based method, called Clustering LAR Applications (CLARA) was developed by Kaufman and Rousseeuw (1990). CLARA draws multiple samples of the data set, applies PAM on each sample, and returns its best clustering as the output. The complexity of each iteration now becomes $O(ks^2 + k(n\text{-}k))$, where s is the size of the sample, k the number of clusters, and n the total number of objects.

The effectiveness of CLARA depends on the sampling method and the sample size. CLARA cannot find the best clustering if any sampled medoid is not among the best *k* medoids. Therefore, it is difficult to determine the sample size. Experiments reported in [Kaufman and Rousseeuw, 1990] indicate that 5 samples of size 40 + 2 *k* gave satisfactory results. However, this is only valid for a small *k*. In the case of road network extraction, we will have hundreds of medoids (i.e., k>100). To assure the quality, the "optimal" sample size has to be about 10 times of *k*, which will require too much computational time for performing PAM on each sample set.

2.3 CLARANS Algorithm

To improve the quality and scalability of CLARA, another clustering algorithm called Clustering Large Applications based upon RANdomized Search (CLARANS) was proposed in [Ng and Han, 1994]. When searching for a better centre, CLARANS tries to find a better solution by randomly choosing object from the other (n-k) objects. If no better solution is found after a certain number of attempts, the local optimal is assumed to be reached. CLARANS has been experimentally shown to be more efficient than both PAM and CLARA. However, its computational complexity is still about $O(n^2)$ [Han *et al*, 2001], where n is the number of objects. Furthermore, its clustering quality is depending on the two predefined parameters: the maximum number of neighbors examined (*maxneighbor*) and the number of local minima obtained (*numloc*).

In summary, in terms of efficiency and quality, none of the existing *k*-medoids algorithms is qualified for spatial clustering for road network extraction. We will propose a new efficient *k*-medoids which can overcome part of the computational problems and is suitable for clustering large data sets.

3 *CLATIN: An Efficient k-Medoids Clustering Method*

When we take a close look at the PAM algorithm, it is found from **Fig.1** that the replacement of the current medoid O_i with the non-medoid object O_h will only affect the small portion of the data set, i.e., the right bottom portion and the left top portion. Therefore, in Step 2 of the PAM clustering, to calculate the total cost of this replacement, we only need to take into account this small portion. The question remaining is then: how to determine efficiently this subset of the original data set?. Thankfully, we can use the triangular network of the medoids to help us find the subset (see **Fig. 2**). This leads us to develop a new efficient *k*-medoids clustering algorithm. We call it Clustering Large Applications with Triangular Irregular Network (CLATIN).

The main steps of our CLATIN can be described as follows:

1. Initialization.
   1) Select *k* representative objects arbitrarily as initial medoids.
   2) Construct the TIN of these *k* medoids.
2. Compute total cost TC_{ih} for all pairs of objects O_i, O_h where O_i is currently selected medoid, and O_h is one of the non-medoid object.
   1) Determine the affected object subset S through a link analysis in the medoid-TIN.

   2) Calculate the total cost TC_{ih} over the neighboring object subset S.

3. Select the pair O_i, O_h which corresponds to $min_{\,Oi,\,Oh}\,(TC_{ih})$. If the minimum TC_{ih} is negative,
   1) replace O_i with O_h,
   2) update the TIN and clustering results locally,
   3) go back to Step 2.
4. Otherwise, for each non-selected object, find the most similar representative object. Halt.

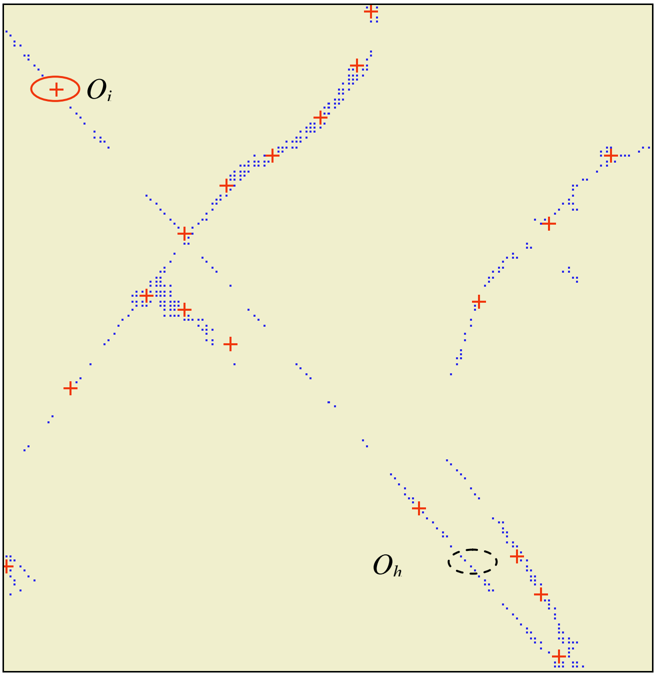

Fig. 1. Illustration of the replacement of current medoid O_i with a non-medoid O_h in a PAM Clustering: Red crosses are current selected medoids, blue points are non-medoids

If we compare the new algorithm with the previous one, there are three additional steps: Step 1.1, Step 2.1 and Step 3.2. These three steps consume a small amount of computational time. In Step 1.1, the TIN can be constructed with a complexity of O($k$log k), where k is the number of nodes; in our case, the number of clusters. Step 2.1 is almost of linear complexity. Step 3.2 is also of linear complexity because we can update the TIN and cluster locally. So in overall, the computation complexity is O(k (n-k) s), where s is the average number of the affected neighbors, which is around 1/10 of n.

There are many existing algorithms that can be chosen for the construction of the TIN of medoids. For example there is the incremental insertion algorithm of Lawson (1977), the divide-and-conquer algorithm of Lee and Schachter (1980), or the planesweep algorithm of Fortune (1987). In our current implementation, we use the incremental insertion algorithm of Lawson (1977), which is also described in [Bourke, 1989]. This algorithm is not worst-case optimal as it achieves O(n^2) time complexity. However, its expected behavior is much better. In average, O(n log n) time complexity is achieved.

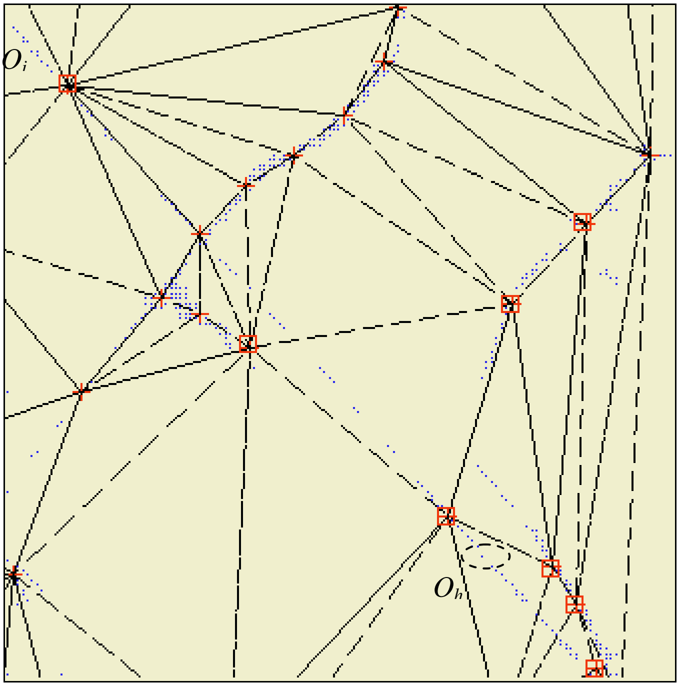

Fig. 2. Illustration of the replacement of current medoid O_i with a non-medoid O_h in a CLATIN clustering: Red crosses are current selected medoids, blue points are non-medoid points, black dash lines are the TIN of current medoids, red rectangles are the possible affected medoids

Table 1 shows the computation time comparison between the new CLATIN approach and other existing k-medoids method on different data sets. The tests were performed in a PC with P4 2.02GHz CPU and 512 MB RAM. For CLARA algorithm, we set the number of sample set to 5 and the size of each sample set is 10*k, where k is the number of clusters.

It can be seen that the CLATIN is faster than the basic PAM in all the cases, while retaining exactly the same clustering quality. The larger the number of clusters, the more time saving is achieved (see Fig 3). CLATIN is faster than the CLARA in the cases of small average cluster population (e.g. less than 50 objects per cluster) only (see Fig. 4). This is not surprising because we fixed the sample size as 10 times of the number of clusters for all the cases, and the computation time of CLARA is less sensitive to the number of objects. However, the clustering quality from CLARA usually is not comparable to that of PAM. To improve the clustering quality, we have

Table 1. Computation time comparison

Case	#Sample	#Cluster	Average population	Computation time (seconds)		
				PAM	CLARA	CLATIN
1	1353	9	150.33	13	<1	11
2	3531	9	392.33	111	1	111
3	6000	9	666.67	351	1	364
4	1353	42	32.21	57	27	29
5	3531	42	84.07	411	29	184
6	6000	42	142.86	1,447	35	633
7	3531	120	29.43	1,100	693	229
8	6000	120	50.00	3,251	675	900
9	13531	120	112.76	18,742	707	3,311
10	13531	270	50.11	45,207	7,424	4,261
11	12859	270	47.63	38,778	6,982	2,685
12	10228	270	37.88	25,165	5,780	2,438

to either increase the number of sample sets or increase the size of each sample set. Both cases will result in a dramatic reduction in computation efficiency. This makes it justifiable to choose CLATIN over CLARA for large applications (e.g. spatial clustering for road network extraction).

4 Applications

In this paper, a similar methodology to the SORM proposed by Doucette *et al* (2001) is used. However, we use the new efficient *k*-medoids clustering algorithm: CLATIN.

The proposed approach starts with an image segmentation using spectral clustering techniques. The road cluster is identified automatically using a fuzzy classification based on a set of predefined membership functions for different landscape types (e.g.

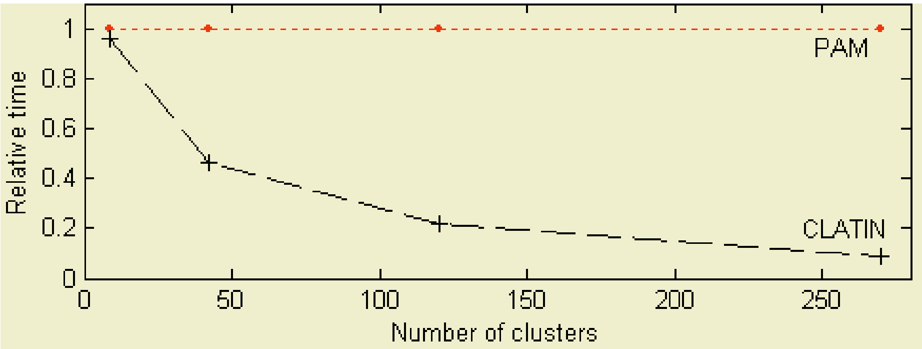

Fig. 3. Comparison of computation time: CLATIN (black dash line) vs. PAM (red dotted)

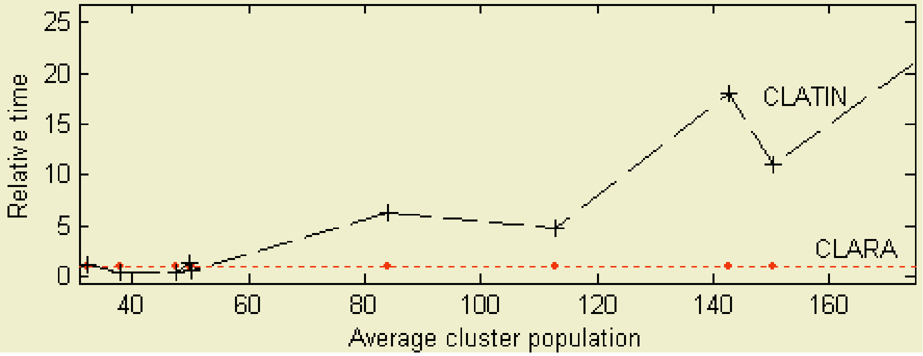

Fig. 4. Comparison of computation time: CLATIN (black dashed) vs. CLARA (red dotted)

water bodies, grasslands, roads). These membership functions are established based n the general spectral signature of each landscape type as well as the characteristics of the image scene under consideration. An angular texture index is used to further reduce the misclassifications between roads, buildings and parking lots. The whole road network is finally extracted by performing a spatial clustering on the road pixels. In this step, the new clustering algorithm is used for spatial clustering purpose and on the road pixels only. The neighboring cluster centers will be linked together to create the corresponding road centerlines and road junctions and to form the road network topology.

The proposed clustering approach for road network extraction has been tested on several IKONOS MS images (4m spatial resolution) covering

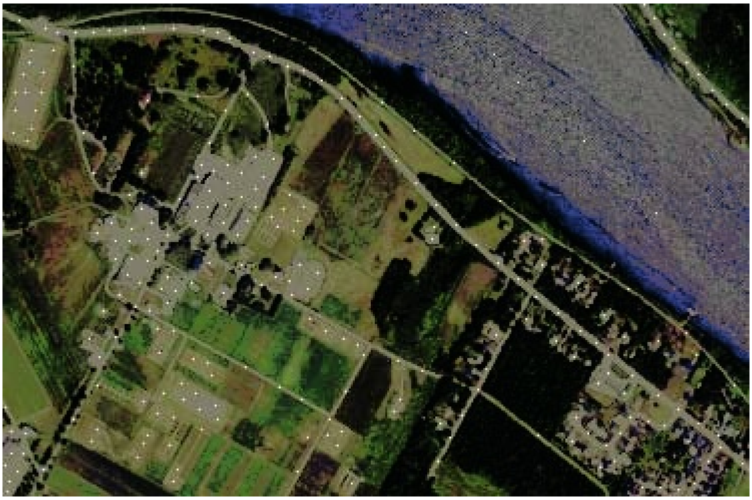

Fig. 5. Road center pixels (white) found by CLATIN algorithm (Test area 1)

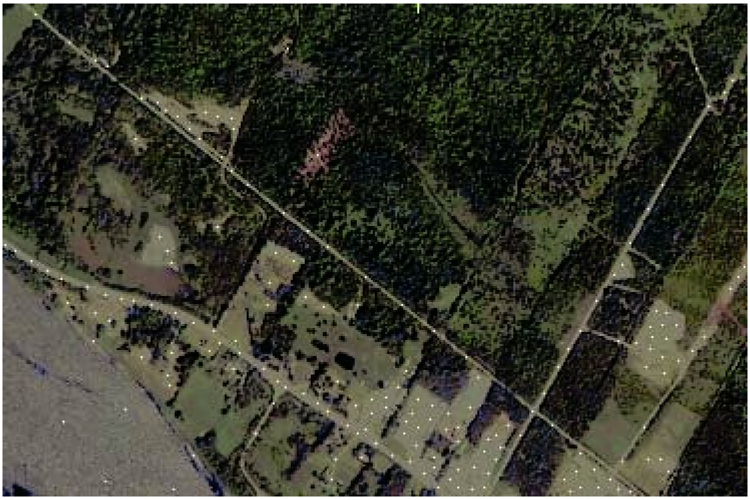

Fig. 6. Road center pixels (white) found by CLATIN algorithm (Test area 2)

the suburban area of the city of Fredericton, Canada. Some results are shown in Fig. 5 and Fig.6. The results are quite encouraging in the sense that the topology of the whole road network has been precisely captured by the resultant road center pixels (in white in Fig 5 and 6). Note that in our current implementation, there is still a large misclassification among roads, buildings and parking lots which makes the results not that pleasing. However, with the introduction of angular texture index and thus the improvement in image classification, we expect much better results from spatial clustering.

5 Conclusions

Spatial clustering has been a useful tool for Geographical Data Mining. Experiments have also demonstrated that spatial clustering approach for linear feature extraction from remotely-sensed imager has some promising advantages over other existing methodologies because it is independent from a conventional edge definition, and can meaningfully exploit multispectral imagery.

K-medoids methods are very robust to the existence of outliers. This makes it an ideal spatial clustering technique for road network extraction from classified imagery. To overcome the computational issue of existing *k*-medoids methods, we have introduced a new *k*-medoids algorithm, which is much faster than the basic PAM algorithm while retaining exactly the same clustering quality. The new CLATIN algorithm is also more efficient than CLARA in the case of large number of clusters.

The future work includes the full implementation of the CLATIN-based road network extraction from multi-spectral imagery, particularly the improvement of image classification accuracy and the implementation of automatic link of clustering centers to form the whole road network. The applicability of clustering approach to road extraction in the urban area will also be one of our next steps.

Acknowledgements

Financial support from the Canadian NCE GEOIDE research program “Automating photogrammetric processing and data fusion of very high resolution satellite imagery with LIDAR, iFSAR and maps for fast, low-cost and precise 3D urban mapping” is much acknowledged.

References

Bourke, P., 1989. Efficient Triangulation Algorithm Suitable for Terrain Modeling, http: //astronomy.swin.edu.au/~pbourke/terrain/triangulate/, accessed on December 10, 2004.

Doucette, P., Agouris, P., Musavi, M., and Stefanidis, A., 1999. Automated Extraction of Linear Features from Aerial Imagery Using Kohonen Learning and GIS Data. In Agouris, P. and Stefanidis, A. (Eds.), 1999. *Integrated Spatial Databases: Digital Images and GIS*, Lecture Notes in Computer Science, Vol. 1737, pp.20-33, Springer-Verlag: Berlin Heidelberg, 1999.

Doucette, P., Agouris, P., Stefanidis, A., Musavi, M., 2001. Self-Organised Clustering for Road Extraction in Classified Imagery. *ISPRS Journal of Photogrammetry & Remote Sensing*, 55, pp.347-358.

Fortune, S. 1987. A Sweepline Algorithm for Voronoï Diagrams. *Algorithmica*, 2(2), pp153-174.

Guo, D., Peuquet, D., and Gahegan, M., 2003. ICEAGE: Interactive Clustering and Exploration of Large and High-dimensional Geodata, *GeoInformatica*, 7(3), pp229 – 253.

Han, J., Kamber, M. and Tung, A., 2001. Spatial clustering methods in data mining: A survey. In *Geographic Data Mining and Knowledge Discovery* [Miller, H.J, and Han, J., Eds]. London: Taylor & Francis Inc.

Kaufman, L. and Rousseeuw, P.J., 1990. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wilsy & Sons.

Lawson, C.L., 1977. Software for C^1 Surface Interpolation. Mathematical Software III (John R. Rice, editor), pages 161-194. Academic Press: New York.

Lee, D.T. and Schachter, B.J., 1980. Two Algorithms for Constructing a Delaunay Triangulation. *International Journal of Computer and Information Sciences*, 9(3), pp219-242.

Ng, R. and J. Han, 1994, Efficient and Effective Clustering Methods for Spatial Data Mining, *Proc. 20th International Conference on Very Large Databases*, Santiago, Chile.