PARAMETER-FREE CLUSTER DETECTION IN SPATIAL DATABASES AND ITS APPLICATION TO TYPIFICATION

Karl-Heinrich Anders[1] **and Monika Sester**[2]
[1] Z/I Imaging GmbH
Oberkochen, Germany
k.anders@ziimaging.de
[2]Institute for Photogrammetry
University of Stuttgart, Germany
monika.sester@ifp.uni-stuttgart.de

KEY WORDS: GIS, Clustering, Spatial Data Interpretation, Data Aggregation, Digital Cartography, Typification

ABSTRACT

The automatic analysis of spatial data sets presumes to have techniques for interpretation and structure recognition. Such procedures are especially needed in GIS and digital cartography in order to automate the time-consuming data update and to generate multi-scale representations of the data. In order to infer higher level information from a more detailed data set, coherent, homogeneous structures in a data set have to be delineated. There are different approaches to tackle this problem, e.g. model based interpretation, rule based aggregation or clustering procedures. In the paper, a parameter-free graph-based clustering approach and an application in the domain of cartography, namely typification is presented. Typification is a generalization operation needed in order to present a set of objects by a subset of representatives. In this way, a collection of objects can be represented by fewer objects in a symbolic representation. An important prerequisite for the legibility of detailed representation is that the structure is preserved. This implies that object clusters are preserved.

1 INTRODUCTION

The ever increasing amount of data and information available demands for an automation of its use. Users need adequate search tools in order to quickly access and filter relevant information. Data Mining has evolved as a branch of computer science, which tries to structure data and find inherent, possibly important, relations in the data. In general, it deals with finding facts by inference; finding information in unstructured data, or in data which is not structured explicitly for the required purpose. The basic tools of Data Mining are machine learning techniques, cluster analysis and interpretation procedures.

In GIS and digital cartography, respectively, there is a growing demand for such techniques: huge spatial data sets are being acquired and have to be kept up to date at ever increasing cycles; furthermore, information of different levels of detail is required in order to compensate for the requirements of different applications. One important application is the scale dependent data representation for quick visualization on a computer screen. In cartography, typically the data of different scales are acquired, managed and updated separately – a highly time consuming and labor intensive task. In order to accelerate update cycles and deliver actual information on-the-fly, tools and techniques for automation of initial data capture and update are required.

An earlier approach (Anders and Sester, 1997) focused on the modeling of the spatial situation in a semantic network and an explicit provision of a set of rules of how to aggregate objects. The prerequisite is the availability of an explicit and complete model of the situation and of the aggregation rules. Such rules are often hard to find and usually also subjective. The aim of this paper is to consider the problem as a general task of finding higher level structures in a seemingly arbitrary collection of (labeled) objects. This can be transferred to the abstract problem of considering the objects and their behavior as a stochastic point process. In this point-collection, meaningful structures have to be identified, namely homogeneous clusters. Thus the approach relies on physiological observations of humans: humans use spatial neighborhood relations in order to find gestalt objects and separate objects from background.

Homogeneity here is considered both concerning geometry, i.e. point density, and concerning semantics, i.e. thematic 'density', namely similarity. Ideally, data mining approaches do not rely on any prior information, e.g. thresholds or parameters, which tune the process. In cluster analysis, usually the number of clusters or an information about the statistical distribution of the data is required. This approach focuses on procedures which are most generally applicable (independent on the type of objects) and need no or only few parameters. Furthermore it is important that arbitrary cluster forms can be identified, when no prior knowledge about the objects is assumed to be known. Such tasks can be tackled by clustering processes – the important prerequisite is the modeling of the neighborhood, which can be achieved by neighborhood graphs.

2 RELATED WORK

In the context of data aggregation, there are many approaches in GIS and in digital cartography, namely in model or database generalization. (Richardson, 1996) and (van Smaalen, 1996) present approaches to come from one detailed scale to the next based on a set of rules. If such rules are known or models of the situation are available, good results can be achieved (cf. (Sester et al., 1998)). However, the main problem being the definition of the rules and the control strategy to infer new data from it (Ruas and Lagrange, 1995). Current concepts try to integrate learning techniques for the derivation of the necessary knowledge (Plazanet et al., 1998), (Sester, 1999).

Clustering is a common descriptive task where one seeks to identify a finite set of categories or clusters to describe the data (Titterington et al., 1985), (Jain and Dubes, 1988). In general one can divide the clustering algorithms in two groups: The *Non-hierarchical* Schemes and the *Hierarchical* Schemes. Existing clustering algorithms, such as ISODATA (Ball and Hall, 1965) k-means (MacQueen, 1967), (Jain and Dubes, 1988), PAM (Kaufman and Rousseeuw, 1990), CLARANS (Ng and Han, 1994), DBSCAN (Ester et al., 1996), CURE (Guha et al., 1998), and ROCK (Guha et al., 1999) are designed to find clusters that fit some static models. For example, k-means, PAM, and CLARANS assume that clusters are hyper-ellipsoidal or hyper-spherical and are of similar sizes. The DBSCAN algorithm assumes that all points of a cluster are *density reachable* (Ester et al., 1996) and points belonging to different clusters are not. All these algorithms can breakdown if the choice of parameters in the static model is incorrect with regarding to the data set being clustered, or the model did not capture the characteristics of the clusters (e.g. shapes, sizes, densities). More information about clustering methods can be found in (Karypis et al., 1999).

3 GRAPH-BASED CLUSTERING

The most powerful methods of clustering in difficult problems, which give results having the best agreement with human performance, are the graph-based methods (Jaromczyk and Toussaint, 1992). The idea is extremely simple: Compute a neighborhood graph (such as the minimal spanning tree) of the original points, then delete any edge in the graph that is much longer (according to some criterion) than its neighbors. The result is a forest and each tree in the forest represents a cluster.

In general, hierarchical cluster algorithms work implicitly or explicitly on a similarity matrix such that every element of the matrix represents the similarity between two elements. In each step of the algorithm the similarity matrix is updated to reflect the revised similarities. Basically, all these algorithms can be distinguished based on their definition of similarity and how they update the similarity matrix. In spatial clustering algorithms one can discriminate between *spatial similarity* and *semantic similarity* which means the similarity of non-spatial attributes.

Spatial Similarity implies the definition of a neighborhood concept which can be defined on geometric attributes, such as coordinate, distance, density, and shape. The computation of a spatial similarity matrix can be seen as the construction of a weighted graph, so called *neighborhood graph*, where each element is represented by a node and each neighborhood relationship (similarity) is an edge. There are efficient algorithms to compute neighborhood graphs (Jaromczyk and Toussaint, 1992) which can be used to compute a spatial similarity matrix.

3.1 Neighborhood Graphs

A general introduction to the subject of Neighborhood graphs is given in (Jaromczyk and Toussaint, 1992). Neighborhood graphs also called *proximity graphs* (Toussaint, 1991), are used as tools in disciplines where shape and structure of point sets are of primary interest. These include for example visual perception, computer vision and pattern recognition, cartography and geography, and biology.

Neighborhood graphs capture proximity between points by connecting nearby points with a graph edge. The many possible notions of *nearby* (in several metrics) lead to a variety of related graphs. It is easiest to view the graphs as connecting points only when certain regions of space are empty. In the following definitions of proximity graphs we will use these notations:

L_p : The distance metric L_p defined as $\delta_p(x,y) = (\sum_{i=1}^{d} |x_i - y_i|^p)^{1/p}$.

$\delta_p(x,y)$: The distance between two points x and y using the metric L_p.

$Ball_p(x,r)$: The open $Ball_p(x,r) = \{y | \delta_p(x,y) < r\}$.

$Lune_{\beta_p}(x,y)$: $Lune_{\beta_p}(x,y) = Ball_p(x(1-\frac{\beta}{2}) + y\frac{\beta}{2}, \frac{\beta}{2}\delta_p(x,y)) \cap Ball_p(x\frac{\beta}{2} + y(1-\frac{\beta}{2}), \frac{\beta}{2}\delta_p(x,y))$.

V : A set of n points in R^d.

$Edge(x,y)$: The vertices x and y have a common edge.

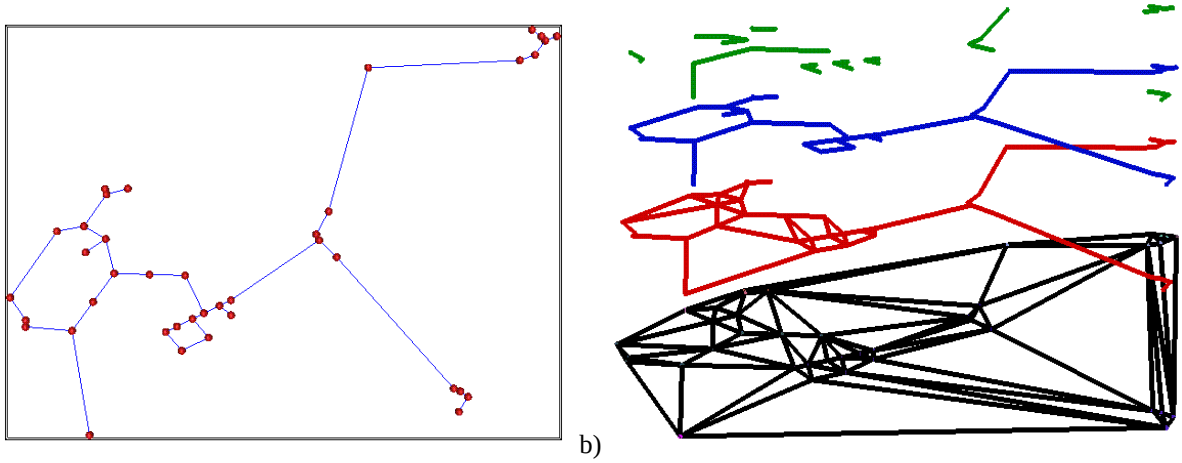
Given a metric L_p some well known proximity graphs are:

- The delaunay triangulation ($DT_p(V)$).
- The nearest neighbor graph (Jarvis and Patrick, 1973),
  $NNG_p(V) = \{Edge(x,y) | x,y \in V \wedge Ball_p(x, \delta_p(x,y)) \cap V = \emptyset\}$.
- The minimum spanning tree ($MST_p(V)$).
- The relative neighborhood graph (figure 1a)) (Toussaint, 1980),
  $RNG_p(V) = \{Edge(x,y) | x,y \in V \wedge Lune_{2_p}(x,y) \cap V = \emptyset\}$.
- The gabriel graph (Gabriel and Sokal, 1969),
  $GG_p(V) = \{Edge(x,y) | x,y \in V \wedge Lune_{1_p}(x,y) \cap V = \emptyset\}$.
- The β-skeleton (Kirkpatrick and Radke, 1985),
  $G_{\beta_p}(V) = \{Edge(x,y) | x,y \in V \wedge Lune_{\beta_p}(x,y) \cap V = \emptyset\}$.
- The sphere of influence graph (Toussaint, 1988),
  $SIG(V) = \{Edge(x,y) | x,y \in V \wedge Ball_p(x, \delta_p(x, NN(x))) \cap Ball_p(y, \delta_p(y, NN(y))) \cap V = \emptyset\}$.
- The α-graphs (Edelsbrunner et al., 1983).

The important relationship between some proximity graphs is that they build a part of hierarchy. Given a point set V and a metric L_p, then for any $\beta \in [1,2]$ the following hierarchy is valid:

$$NNG_p(V) \subseteq MST_p(V) \subseteq RNG_p(V) \subseteq G_{\beta_p} \subseteq GG_p(V) \subseteq DT_p(V).$$

In figure 1b) the hierarchical relationship between the Nearest Neighbor Graph, the Relative Neighborhood Graph, the Gabriel Graph, and the Delaunay Triangulation of a point set is shown.



a) b)

Figure 1: a) RNG of a point set. b) Hierarchical relationship between proximity graphs (top down: NNG, RNG, GG, DT)

The computation of such a hierarchy needs $O(nlogn)$ time, because the computation of the Delaunay Triangulation needs $O(nlogn)$ time and any subgraph can be computed from its supergraph in $O(n)$ time. For example, an algorithm for the RNG in the Euclidean metric using the Delaunay Triangulation was developed by (Supowit, 1983).

4 HIERARCHICAL GRAPHBASED CLUSTERING

In our approach we use the hierarchical relationship between proximity graphs to represent a near to a far neighborhood model. Our algorithm can be described as follows:

The first basic step is the computation of the Delaunay Triangulation (DT) from a given set of points. In the next step we compute first the Gabriel Graph (GG) from the DT, second the Relative Neighborhood Graph (RNG) from the GG, and third the Nearest Neighbor Graph (NNG) from the RNG (figure 1a)). Then we activate the edges of the NNG to start with the nearest neighbor model. Then all given points (graph nodes) are initialized as a single cluster. Every cluster contains a set of *inner* edges and a set of *outer* edges. The inner edges connect nodes which belongs to the same cluster and the outer edges connect nodes which belongs to different clusters. Every cluster is characterized by the median of the inner edge sizes (*cluster density*) and the *cluster variance*. The cluster variance is the median deviation of all inner and outer edge sizes from the cluster density. Using the inner and outer edges to compute the variance introduce an uncertainty factor to our model. At the beginning every initial cluster has no inner edges and therefore a density of zero, but the variance will be none zero, because every node in the NNG belongs at least to one edge. All initial clusters are put into a priority queue, ordered by their density and variance values. The first cluster in the priority queue is selected (cluster with the highest density) and merged with all of his *valid* neighbor clusters. Valid neighbor clusters are clusters which are connected by an outer edge and meet the following heuristic constraints:

- *Density compatibility* of two clusters X, Y:
  $Min(X) \leq Median(Y) \leq Max(X) \wedge Min(Y) \leq Median(X) \leq Max(Y)$.
- *Distance compatibility* of two clusters X, Y:
  $Max(X \cup Y) \leq Max(X) \wedge Max(X \cup Y) \leq Max(Y) \wedge Min(X) \leq Min(X \cup Y) \wedge Min(Y) \leq Min(X \cup Y)$.
- *Variance compatibility* of two clusters X, Y:
  $Variance(X \cup Y) \leq Variance(X) \vee Variance(X \cup Y) \leq Variance(Y)$.

After the merging all valid neighbor clusters are removed from the priority queue. Then repeat the selecting and merging step until no more clusters with valid neighbors can be found. The result are the clusters based on the NNG. In the next step the RNG edges are activated an the same procedure as for the NNG is repeated. Then the GG edges are activated and finaly the edges of the DT are processed.

One basic aim of our approach was to detected building clusters for map generalization (see next chapter). Figure 2 a) and b) shows the clustering result of two 2D point sets (centroids) derived from 2D building groundplans.

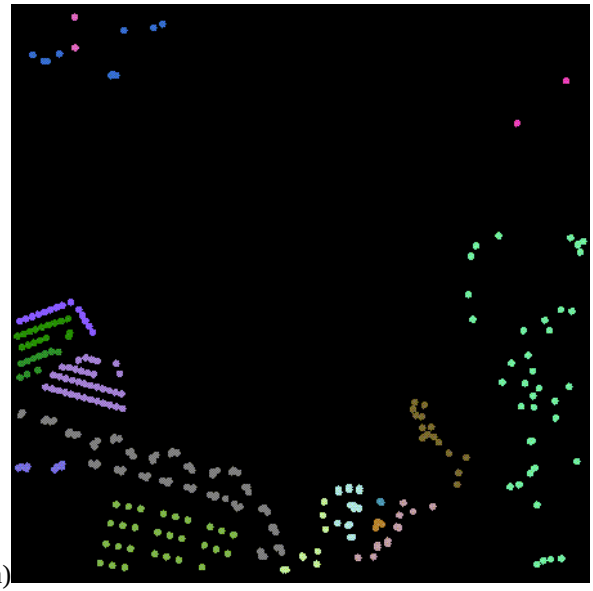
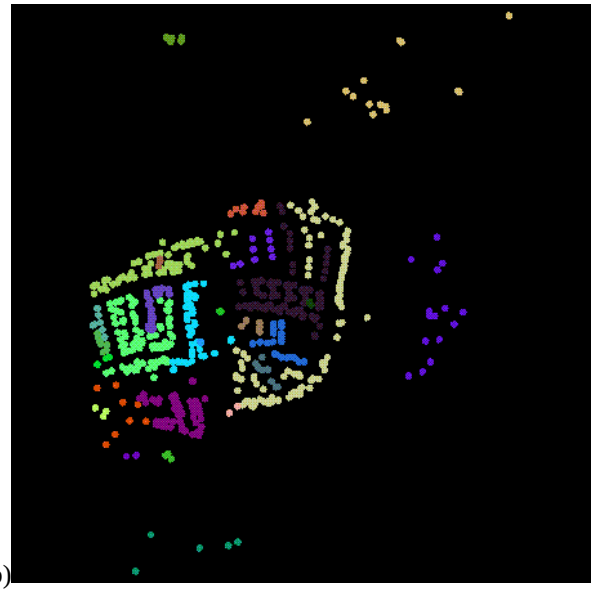
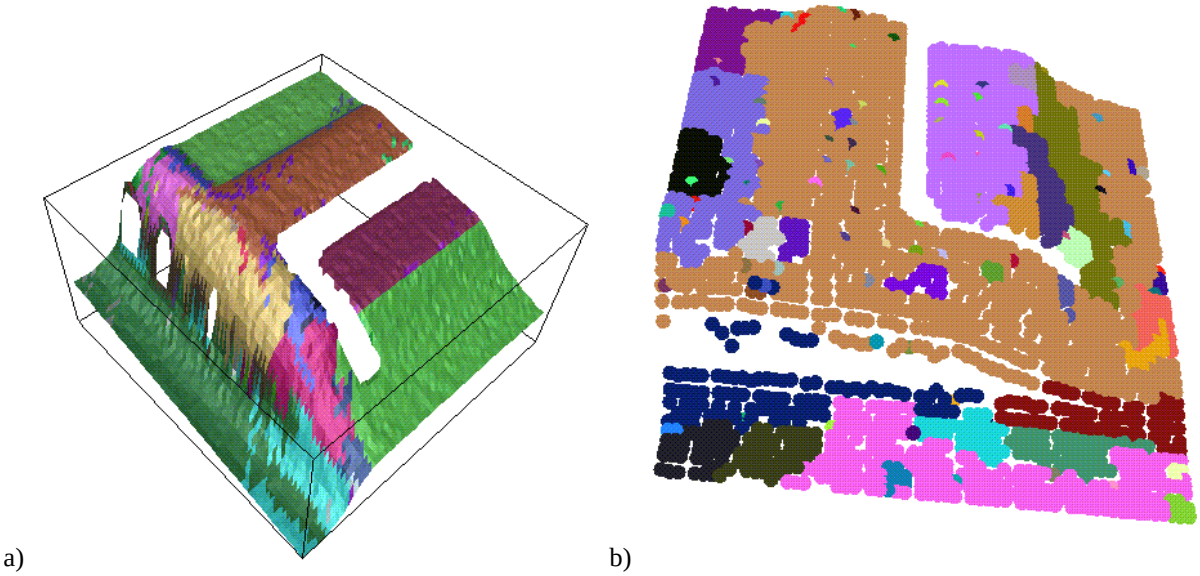
a)  b) 

Figure 2:

We applied our clustering method also to a measured 3D object point cloud. Figure 3a) shows the result of a special segmentation method using a surface model, the surface curvature, and requires some user-defined parameters. Figure 3b) shows the result of our clustering process without any user-defined parameters using only the given 3D points.



a) b)

Figure 3: Segmentation of a 3D point cloud. a) Reference segmentation. b) Graphbased result.

Our approach can also be applied to images. Every image pixel is transformed to a 3D point represented by row, column, and gray value. Figure 4 shows the result of an example image.

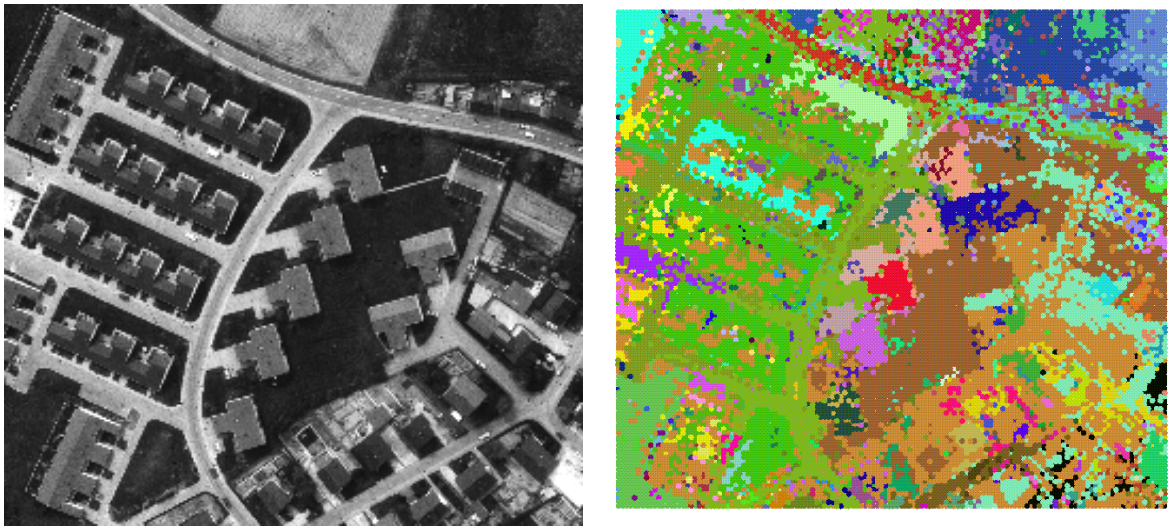


Figure 4: Image classification. Left: Gray value image. Right: Clustering result.

5 CLUSTER DETECTION AS A PREREQUISITE FOR TYPIFICATION

Generalization is needed in order to limit the amount of information on a map by enhancing the important information and dropping the unimportant one. Triggers for generalization are on the one hand limited space to present all the information; on the other hand but also the fact that different scales of an object are needed in order to reveal its internal structure.

Typification is a generalization operation that replaces a large number of similar objects by a small number – while ensuring that the typical spatial structure of the objects is preserved. Consider e.g. a set of lakes in Finland: when looking at this spatial situation at a different scale or resolution, the typical distribution of the lakes should still be preserved. The same holds for buildings in a city: different parts of the city exhibit different cluster densities. These differences have to be preserved, if not enhanced, by typification. There are some approaches dealing with different kinds of objects: (Mller and Wang, 1992) use mathematical morphology to typify natural areal objects. Their principle is to enhance big objects and reduce small ones – unless they are important. Typification for linear structures is proposed by (Regnauld, 1996). Based on a Minimum Spanning Tree clustering groups are detected; then the relevant objects within these groups are replaced

by typical exemplars. This approach for building typification is motivated by the phenomenological property of buildings being aligned along streets – thus a one-dimensional approach is feasible.

Our approach is similar, however tackles the two dimensionsional problem. The above described clustering is applied to buildings, thus delineating buildings clusters, and their respective densities (or mean distances, respectively). After clustering, the number of objects within the clusters has to be reduced. The reduction factor can e.g. be derived using the black-and-white–ratio, which is to be preserved before and after generalization, or Tpfer's radical law. The problem now is to decide *which* object has to be removed. This question is decisive, since the removal of one object results in gaps. Therefore a solution must be sought to preserve the initial cluster density (the mean distances between the objects) after the elimination of the objects.

This is achieved using the following strategy: after the elimination of the objects, all the objects inside the cluster are rearranged in order to reflect the original density. This strategy is implemented in a displacement framework PUSH, described in these proceedings (Sester, 2000). This program allows for displacement of geometric primitives in order to ensure minimal distances between the objects. In order to displace cluster elements, the distances between the objects within a cluster are set to be the original distances, which are computed in the clustering process. Thus intra-cluster distances are set to the respective a priori distances, whereas inter-cluster distances are set to the required distances in the displacement process. This approach is visualized in the following example. The situation in a building block is too dense to be displayed in the reduced scale 5, a). Thus it has to be typified – preserving the original structure. The application of the clustering yields the result given in figure 5, b), where each color indicates objects of the same cluster. Obviously 5 clusters have been detected.

Then, one of the objects in the most crowded cluster is eliminated (5, c). Applying the above described displacement yields the result, that the gap is closed and the objects are displaced to their original spacing 5,d).

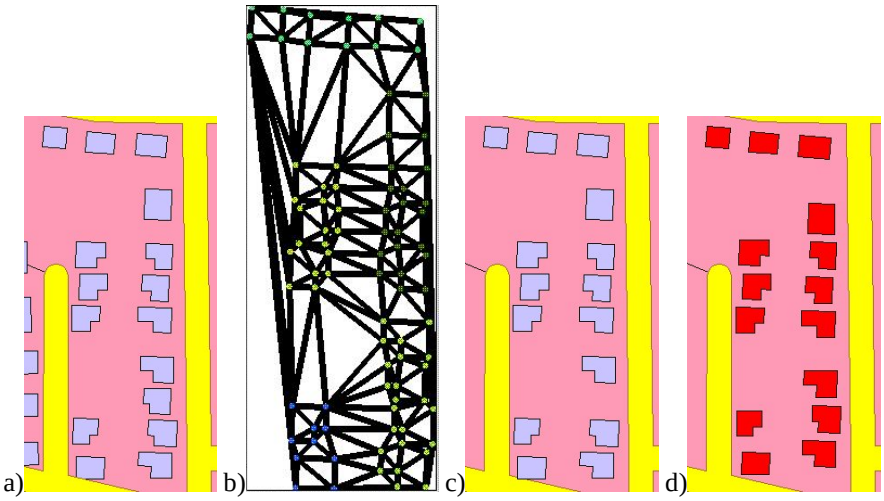


Figure 5: Elimination of one object in the bottom right cluster (left); result of the displacement of all the objects in the scene(middle) (right).

The important property of the approach is that the selection of the object to be eliminated is arbitrary – as the original spacing is enforced, the objects will be placed in the correct manner. This is shown in figure 6. The remaining three possibilities for removing an object from the cluster have been calculated: although the position of the cluster slightly varies, still the cluster shape is preserved.

The examples showed how linear clusters have been detected and typified. The approach, however is general and can be applied to non-linear situations as well. If the objects inside the clusters are placed randomly, then the above sketched procedure is very useful. Problem occur, however, if there is a regular structure (e.g. a regular grid) within the clusters – eliminating an object causes the remaining objects move into the gaps, and thus destroying the regular structure.

6 CONCLUSION

In this work we have described a general approach to detect clusters in point sets without using any user defined parameters. We have shown that this cluster method can be used as a preprocessing step for the typification in cartographic generalization. Intended to find clusters in settlement data this approach can also be used for the segmentation of image and 3D measurement data.

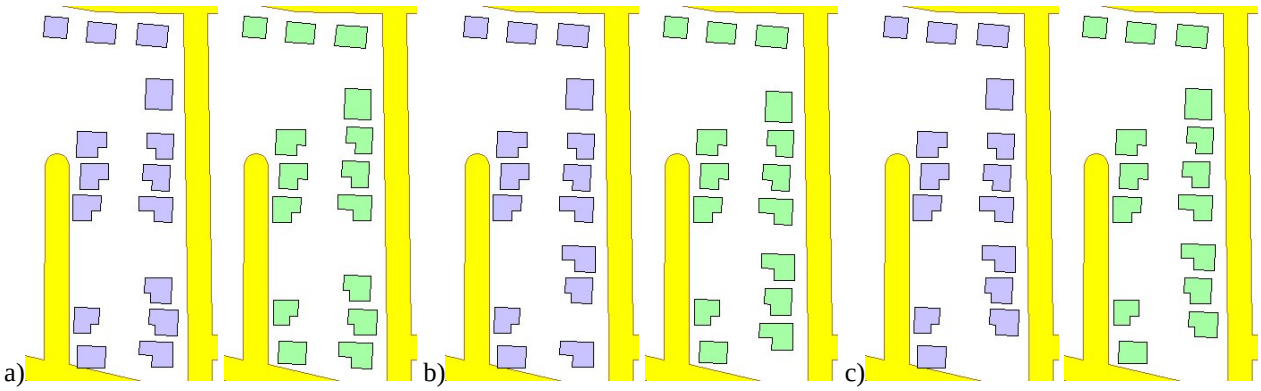


Figure 6: a) Elimination of upper object and result of displacement. b) Elimination of middle object. c) Elimination of lower object.

Our approach use only *metrical scales* for the computations, but further work has to be done to include so called *nominal scales* in the clustering process to work also on non numerical values. Further work has also to be done on the computation of the characteristics of the found clusters. These characteristics are geometric features, like size, shape, average density, etc. In addition to these unary features, also binary features – relations – between the clusters can be used. These characteristics can then help either to identify clusters with similar characteristics, or even to identify objects. This can be achieved by well known pattern recognition or interpretation procedures, e.g. model based interpretation.

Especially the computation of the *shape* of a point set is a non-trivial problem. In contrast to other geometric notions, such as diameter, volume, or convex hull the geometric notion of shape has no associated formal meaning (Edelsbrunner and Mücke, 1994). A fair amount of related work has been done for planar point sets, and some for three dimensional point sets. One of the first who considered the problem of computing the shape of a point set as a generalization of the convex hull was (Jarvis, 1977). A suitable method to describe the shape of point sets are the so called α*-shapes*. A general and mathematically well defined concept of shapei ntroduced by (Edelsbrunner et al., 1983), (Edelsbrunner and Mücke, 1994).

REFERENCES

Anders, K.-H. and Sester, M., 1997. Methods of data base interpretation - applied to model generalization from large to medium scale. In: W. Förstner and L. Plümer (eds), SMATI '97: Semantic Modelling for the Acquisition of Topographic Information from Images and Maps, Birkhäuser, pp. 89–103.

Ball, G. and Hall, D., 1965. Isodata: a novel method of data analysis and pattern classification. Technical Report AD 699616, Stanford Research Institute.

Edelsbrunner, H. and Mücke, E., 1994. Three-dimensional alpha shapes. In: ACM Transactions on Graphics, Vol. 13number 1, pp. 43–72.

Edelsbrunner, H., Kirkpatrick, D. and Seidel, R., 1983. On the shape of a set of points in the plane. In: IEEE Transactions on Information Theory, Vol. IT-29, pp. 551–559.

Ester, M., Kriegel, H.-P., Sander, J. and Xu, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd. International Conference on Knowledge Discovery and Data Mining (KDD-96).

Gabriel, K. and Sokal, R., 1969. A new statistical approach to geographic variation analysis. In: Systematic Zoology, Vol. 18, pp. 259–278.

Guha, S., Rastogi, R. and Shim, K., 1998. Cure: An efficient clustering algorithm for large databases. In: Proc. of 1998 ACM-SIGMOD International Conference on Management of Data.

Guha, S., Rastogi, R. and Shim, K., 1999. Rock: A robust clustering algorithm for categorical attributes. In: Proc. of the 15th International Conference on Data Engineering.

Jain, A. and Dubes, R., 1988. Algorithms for Clustering Data. Prentice Hall.

Jaromczyk, J. and Toussaint, G., 1992. Relative neighborhood graphs and their relatives. In: Proceedings IEEE, Vol. 80number 9, pp. 1502–1517.

Jarvis, R., 1977. Computing the shape hull of points in the plane. In: Proceedings of the IEEE Computing Society Conference on Pattern Recognition and Image Processing, pp. 231–241.

Jarvis, R. and Patrick, E., 1973. Clustering using a similarity measure based on shared near neighbours. In: IEEE Transactions on Computers, Vol. 22number 11, pp. 1025–1034.

Karypis, G., Han, E.-H. S. and Kumar, V., 1999. Chameleon: A hierarchical clustering algorithm using dynamical modeling. To appear in the IEEE Computer or via internet at http://winter.cs.umn.edu/ karypis/publications/data-mining.html.

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

Kirkpatrick, D. and Radke, J., 1985. A framework for computational morphology. In: G. Toussaint (ed.), Computational Geometry, North-Holland, pp. 217–248.

MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations. In: Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp. 281–297.

Mller, J. and Wang, Z., 1992. Area-patch generalisation: a competitive approach. The Cartographic Journal 29, pp. 137–144.

Ng, R. and Han, J., 1994. Efficient and effective clustering method for spatial data mining. In: Proc. of 1994 Int. Conf. on Very Large Data Bases (VLDB'94), Santiago, Chile, pp. 144–155.

Plazanet, C., Bigolin, N. and Ruas, A., 1998. Experiments with learning techniques for spatial model enrichment and line generalization. GeoInformatica 2(4), pp. 315–334.

Regnauld, N., 1996. Recognition of building clusters for generalization. In: M. Kraak and M. Molenaar (eds), Advances in GIS Research, Proc. of 7th Int. Symposium on Spatial Data Handling (SDH), Vol. 1, Faculty of Geod. Engineering, Delft, The Netherlands, pp. 4B.1–4B.14.

Richardson, D., 1996. Automatic processes in database building and subsequent automatic abstractions. Cartographica, Monograph 47 33(1), pp. 41–54.

Ruas, A. and Lagrange, J., 1995. Data and knowledge modelling for generalization. In: J.-C. Müller, J.-P. Lagrange and R. Weibel (eds), GIS and Generalization - Methodology and Practice, Taylor & Francis, pp. 73–90.

Sester, M., 1999. Knowledge acquisition for the automatic interpretation of spatial data. Accepted for Publication in: International Journal of Geographical Information Science.

Sester, M., 2000. Generalization based on least squares adjustment. In: IAPRS, Vol. 33, ISPRS, Amsterdam, Holland.

Sester, M., Anders, K.-H. and Walter, V., 1998. Linking objects of different spatial data sets by integration and aggregation. GeoInformatica 2(4), pp. 335–358.

Supowit, K., 1983. The relative neighborhood graph, with an application to minimum spanning trees. J.Assoc.Comput.Mach. 30, pp. 428–448.

Titterington, D., Smith, A. and Makov, U., 1985. Statistical Analysis of Finite Mixture Distributions. John Wiley and Sons, Chichster, U.K.

Toussaint, G., 1980. The relative neighborhood graph of a finite planar set. In: Pattern Recognition, Vol. 12, pp. 261–268.

Toussaint, G., 1988. A graph-theoretical primal sketch. In: G. Toussaint (ed.), Computational Morphology, North-Holland, pp. 229–260.

Toussaint, G., 1991. Some unsolved problems on proximity graphs. In: D. Dearholt and F. Harary (eds), Proceedings of the First Workshop on Proximity Graphs. Memoranda in Computer and Cognitive Science MCCS-91-224, Computing research laboratory, New Mexico State University, La Cruces.

van Smaalen, J., 1996. Spatial abstraction based on hierarchical re-classification. Cartographica, Monograph 47 33(1), pp. 65–74.