First publ. in: Image Analysis : 14th Scandinavian Conference, SCIA, 2005, Joensuu, Finland, June 19 - 22, 2005; proceedings / Heikki Kalviainen ... (eds.). - Berlin [u.a.] : Springer, 2005. - S. 872-881. - (Lecture notes in computer science ; 3540). - ISBN 978-3-540-26320-3

Clustering Based on Principal Curve

Ioan Cleju[1], Pasi Fränti[2], Xiaolin Wu[3]

[1] University of Konstanz, Department of Computer and Information Science,
Fach M697, 78457 Konstanz, Germany
cleju@inf.uni-konstanz.de
[2] University of Joensuu, Department of Computer Science, P. O. Box 111,
80110 Joensuu, Finland
franti@cs.joensuu fi
[3] McMaster University, Department of Electrical and Computer Engineering,
L8G 4K1, Hamilton, Ontario, Canada
xwu@mail.ece mcmaster.ca

Abstract. Clustering algorithms are intensively used in the image analysis field in compression, segmentation, recognition and other tasks. In this work we present a new approach in clustering vector datasets by finding a good order in the set, and then applying an optimal segmentation algorithm. The algorithm heuristically prolongs the optimal scalar quantization technique to vector space. The data set is sequenced using one-dimensional projection spaces. We show that the principal axis is too rigid to preserve the adjacency of the points. We present a way to refine the order using the minimum weight Hamiltonian path in the data graph. Next we propose to use the principal curve to better model the non-linearity of the data and find a good sequence in the data. The experimental results show that the principal curve based clustering method can be successfully used in cluster analysis.

1 Introduction

Clustering is a general classification procedure that divides a set of objects in different classes. Objects from the same class should be similar to each other. There is no indication about their number but only the properties of the objects in the set. Clustering algorithms are used in a multitude of applications mostly related to pattern matching, image processing, data mining, information retrieval or spatial data analysis. In image analysis, the interest for clustering algorithms comes from tasks as compression, segmentation or recognition. The NP-completeness of the problem [1] motivated the research for good and fast heuristics.

The scalar quantization, a special case of clustering problem, can be optimally solved in linear time [2, 3]. The main difference to vector quantization is that the scalar space is naturally ordered and optimal clusters are always subsequences of the dataset. The main contribution of this work is a new heuristic clustering method, based on the principal curve, which orders the dataset and applies a segmentation algorithm similar to the one used for scalar quantization.

1.1 Problem Definition

The result of the clustering is a *partitioning* of the original set that maps each data point to its class, or cluster. The quality of the clustering is defined by the objective function *f* that assigns a real number to each possible clustering. Thus, the clustering problem refers to finding a clustering that optimizes the objective function. The size of the clustering is defined as the number of different clusters. The *K-clustering* problem is a simplification of the original clustering problem and refers to finding a clustering of a given size *K*.

In this work we will consider only the K-clustering problem, although for simplicity we might refer to it as just the clustering problem. As the objective function, we will use the *mean squared error* (MSE). In this sense, it is of interest to designate to each cluster a unique representative, and assign the value of the representative to the mean vector of the cluster (centroid). The set of representatives for every cluster defines the *codebook*. The elements of the codebook are called *code-vectors*. The error representation for a point is defined as the distance from the point to its corresponding code-vector. The goal is to find a clustering that minimizes the MSE. Notice that on these considerations, a specification for one of partitioning, clustering or codebook will uniquely determine the other two.

1.2 Related Work

The clustering algorithms are very diverse, a good overview can be found in [4]. At the top level, the different approaches can be classified as hierarchical and partitional. Hierarchical algorithms produce a series of partitions, known as dendogram, while partitional ones produce one partition. The dendogram can be constructed top-down, by hierarchical divisive algorithms, or bottom-up by agglomerative clustering algorithms. The most used algorithms in cluster analysis are squared error algorithms, such as K-means [5]. The resulted clustering is highly dependent on the initialization and therefore K-means is usually used to fine-tune a solution given by another algorithm. Graph theoretic algorithms model the data as a graph (e.g. minimum spanning tree) and delete the edges that are too expensive [6]. Mixture resolving approaches assume that the data was generated by an unknown distribution and try to determine its parameters [7]. Fuzzy algorithms [8] and artificial neural networks [9] have been successfully used in clustering. New approaches using genetic algorithms give also good results [10].

1.3 Overview of Our Work

In this work we study different possibilities to reformulate the clustering problem as order constrained clustering [11]. The latter problem can be optimally solved in polynomial time. Section 2 defines order constrained clustering, shows that scalar quantization is a special case of order constrained clustering and explains a method for solving it optimally. A possibility to apply a similar method on vector spaces using the clustering based on principal axis algorithm is described in subsection 2.1.

The approach has been applied in literature [12], but we extend it introducing a sequence tuning method based on minimum weight Hamiltonian path, in subsection 2.2. The main contribution of this work, presented in section 3, describes our new clustering algorithm based on the principal curve. Section 4 presents the results and section 5 provides the conclusions and future development possibilities.

2 Order Constrained Clustering

Order constrained clustering is a special case of clustering [11]. The set is ordered and the clusters are subsequences of the dataset sequence. The problem can be optimally solved in polynomial time [11], however the clustering is usually optimal only in the context of the order constraint.

Scalar quantization can be formulated as order constrained clustering considering the order in the scalar space. Due to the convex hull of the optimal clusters, the solution for order constrained scalar quantization is optimal for the unconstrained problem as well. Scalar quantization can be solved using the *minimum weight K-link path* problem [13]. An oriented graph is constructed for the ordered set, containing edges from any node to all the nodes that appear later in the sequence. The weight of an edge is equal to the distortion of one cluster that contains all the data points between the corresponding nodes. The minimum weight path from the first to the last node, consisting of K edges, corresponds to the optimal clustering of the set. It can be optimally found by a dynamic programming procedure in linear time [2, 3].

The order constrained clustering problem in vector space can be formulated as the minimum weight K-link problem as well. The quality of the result strongly depends on the order relation. Different possibilities to order the dataset will be studied next (see Fig. 1, Fig. 3).

2.1 Clustering Based on Principal Axis

The dataset can be sequenced using the projections over a one-dimensional space (see Fig. 1). The principal axis of the set, computed using the first principal component [14], is a linear space that probably contains the most information, as it maximizes the dispersion of data.

The method has been applied to color quantization in [12]. Principal axis based clustering (PAC) gives good results only if the dispersion of the data along the principal axis is significantly higher than the dispersion in other directions, so that the dataset fits the linear model. It can be observed (see Fig. 2) that the code-vectors are close to the principal axis. The improvement obtained by K-means tuning is considerable but the solution is dependent on the initialization clusters.

2.2 Revising the Order by Hamiltonian Shortest Path

The quality of the order depends on how the spatial relations of the data points are maintained during the projection to the one-dimensional space. The projection on the

principal axis introduces great distortions in preserving the spatial relations. We therefore aim at tuning the order by minimizing the total weight of the sequence path.

We will consider the order given by the principal axis as an approximation for the *minimum weight Hamiltonian path* (MWHP), and try to improve it by simple heuristics (see Fig. 1), as finding the minimum weight Hamiltonian path in a graph is an NP-complete problem [15]. First, we consider short subsequences for which the minimum weight path can be found easily, and optimize the subsequence path length correspondingly. As the size of the subsequences iteratively increases, we do not look for the optimal path but for approximations.

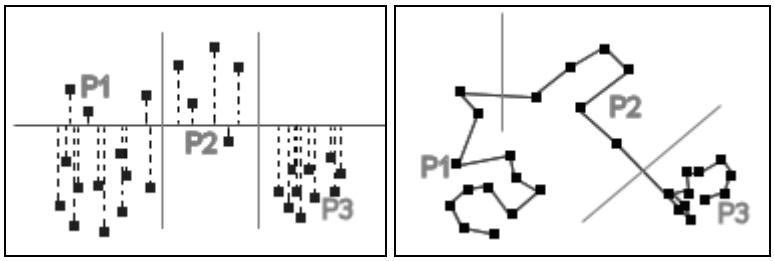

Fig. 1. Examples showing clustering based on principal axis (*left*) and minimum weight Hamiltonian path (*right*)

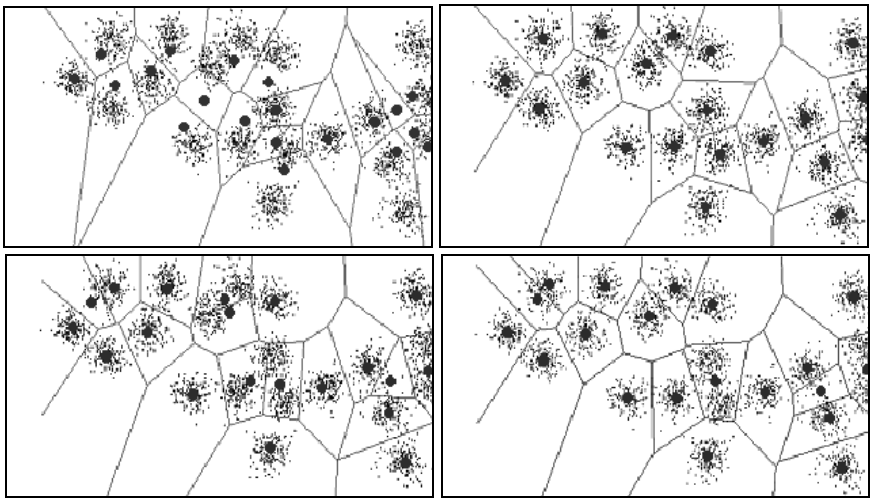

Fig. 2. Results of the algorithms based on principal axis projection: PAC (*upper left*), PAC tuned by K-means (*upper right*), PAC tuned by MWHP (*bottom left*) and PAC tuned by MWHP and K-means (*bottom right*)

The new clustering result shows significant improvement, but after K-means tuning it is not necessarily better (see Fig. 2). The order given by the principal axis assures a good dispersion of code-vectors along its direction; PAC is advantageous to

use if the data dispersion is significant only in principal axis direction. It seems that the best order should be flexible enough to capture the global layout of the clusters but not too detailed, in order to prevent the path jumping between clusters, or going through the same cluster several times.

3 Clustering Based on Principal Curve

As the principal axis is a linear model too rigid to fit the data, we propose to use *principal curves* [16, 17, 18, 19, 20, 21] to order the dataset. The principal curves are one-dimensional spaces that can capture the non-linearity from the data and can better preserve the adjacency of data points. Fig. 3 shows an example of the new method. The main steps are:

- construction of the curve,
- projection over the curve and forming the data sequence,
- order constrained clustering, and
- constructing the Voronoi cells.

There are different definitions for the principal curve. The initial approach intuitively describes it as a smooth one-dimensional curve that passes through the "middle" of data [16]. The curve is self-consistent, smooth and does not intersect itself. An application that uses closed principal curves to model the outlines of ice floes in satellite images is developed in [17]. An improved variant that combines the former approaches is applied to classification and feature extraction [18]. Principal curves are defined in [19] as continuous curves of a given maximal length, which minimize the squared distance to the points of the space. An incremental method similar to K-means for finding principal curves is developed in [20]. Instead of considering the total length of the curve as in [19], the method in [21] constrains the sum of the angles along the curve (the total turn).

We propose to use the *principal curve with length constraint*, as it is defined in [19], to project and order the dataset; from now on we will simply refer to this curve as principal curve. The principal curve minimizes the distortion of the points to the curve. This assures that the adjacency of the points can be preserved on the curve.

The correspondence between the principal curves and Kohonen's self-organizing maps [9] has been pointed out in the literature [22]. However the algorithm for principal curve with length constraint substantially differs from the ones that are based on self-organizing maps; the distances to the data points are measured from the vertices and the segments as well, and not only from the vertices [19].

The practical learning algorithm of the principal curve constructs a sub-optimal polygonal line approximation of the curve. The main difference to the theoretical algorithm is that the length is not provided as a parameter, but the algorithm optimizes it. The procedure is iterative, each step one segment is split and then all the segments of the polygonal line are optimized.

In the segment optimization step, each vertex position is separately optimized, keeping all the other vertexes fixed. A Lagrangian formulation that combines the squared error of the data points that project on adjacent edges and the local curvature is minimized. The smoothing factor that weights the local curvature measure is

heuristically found. It can be controlled by the penalty coefficient. The modification of the penalty coefficient determines the shape and indirectly controls the length of the curve. A small value will determine a very long curve, at the limit just a path in the dataset that has null error. A very large value will determine a shape very similar to the principal axis. The stopping criterion also uses a heuristic test that is controlled by a parameter.

After the set is ordered along the principal curve, dynamic programming is applied and the optimal clustering for the constrained clustering is found. Voronoi cells are formed and K-means can be iterated. As shown in Fig. 4, for a good parameterization of the curve the results are very close to the local optimum.

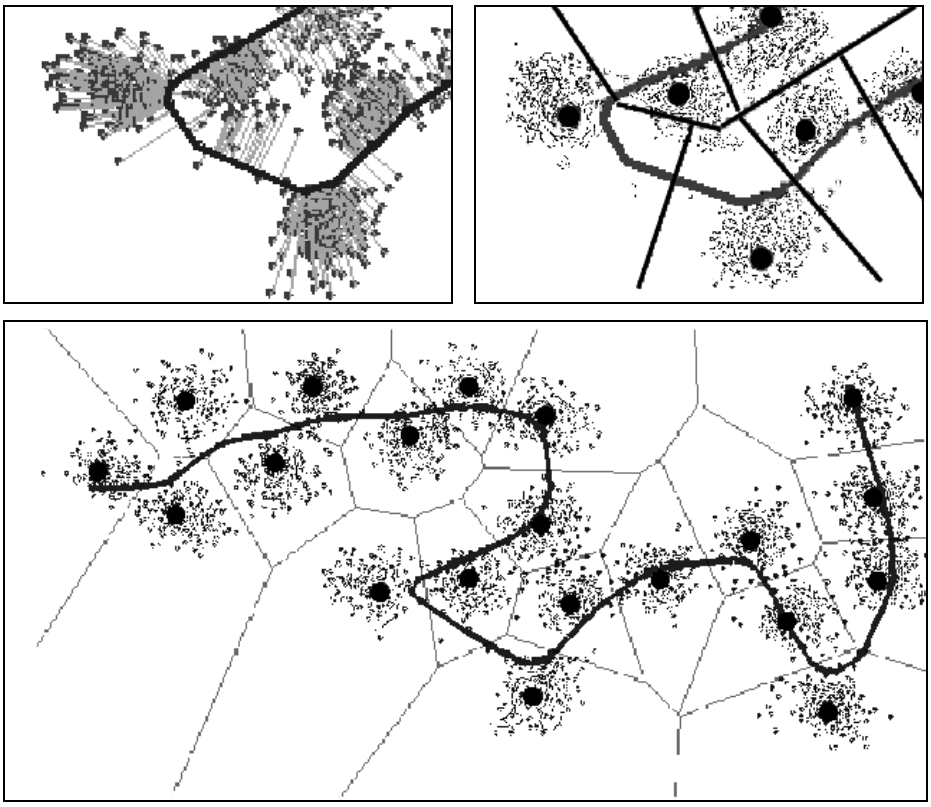

Fig. 3. Projection of data points over the principal curve (*upper left*), optimal segmentation of the sequence (*upper right*) and final clusters (*bottom*)

3.1 Choice of Parameters

Except for the number of clusters, the parameters for the clustering algorithm are used for the curve construction. One parameter influences the stopping criterion and

another one influences the curvature (the penalty coefficient). The parameter that controls the stopping criterion does not significantly influence the clustering result and the value provided in [19] is good for our purpose.

Changes of the penalty coefficient influence the shape of the curve and the clustering (see Fig. 4). Longer curves allow data points from one cluster to project on several regions of the curve, while shorter curves allow different clusters to project on overlapping regions. This coefficient must therefore be tuned depending on the data. Our experiments showed that the value proposed in [19] to 0.13 does not provide the best results in clustering (see Section 4).

3.2 Complexity of the Method

The principal curve algorithm has the complexity of $O(N^{5/3})$. It can be reduced to $O(SN^{4/3})$ if S-segments polygonal approximation of the curve is considered. The overall complexity for clustering is $O(KN^2)$ and it is given by the dynamic programming technique applied to order constrained clustering.

4 Experimental Results

For the experiments we have used three types of datasets. The A datasets (A1, A2, A3) are artificial and contain different numbers of two-dimensional Gaussian clusters having about the same characteristics. The S sets (S1, S2, S3, S4) are two-dimensional artificial datasets containing clusters with varying complexity in terms of spatial data distributions. The real datasets (House, Bridge, Camera, Missa) come from image data, representing color information (House), 4×4 non-overlapping blocks of gray image (Bridge and Camera) and 4×4 difference blocks of two subsequent frames in the video sequence (Missa). Correspondingly, the datasets have 3 and 16 dimensions.

The experiments have been carried out for 15 different values of the penalty coefficient, ranging from 0.001 to 0.22. For the artificial datasets that present clusters in the data, the MSE as a function of penalty coefficient clearly has a minimum. Good values are obtained for the penalty coefficient in the range 0.01 to 0.08 (see Fig. 4). For data that does not have the evidence of clusters, the minimum is not clear and good clustering can be obtained for lower values of the penalty coefficient as well.

The comparative results include K-means and *randomized local search* (RLS) [23]. The K-means MSE values are the best results obtained by 10 repeated trials. The values for the RLS method have been considered when the value of the MSE stabilizes; a slightly better solution is found after a larger number of iterations. The results for clustering based on principal axis (PAC) and principal curve (PCU), with and without the K-means tuned versions, are compared. The MSE value for the principal curve clustering was chosen as the best for the different penalty coefficients. Numerical results are shown in Tables 1, 2 and 3.

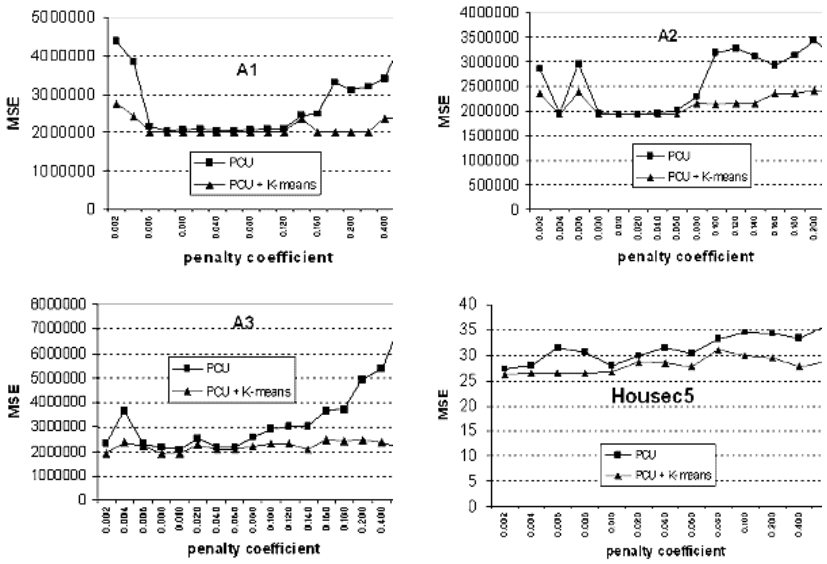

Fig. 4. Results (MSE) for principal curve clustering (PCU) as a function of the penalty coefficient

Table 1. Comparison of the results for the A datasets

Method	A sets		
	A1 ($*10^5$)	A2 ($*10^5$)	A3 ($*10^5$)
K-means	20.24	19.32	19.29
RLS	20.24	19.32	19.29
PAC	83.00	156.57	176.59
PAC + K-means	20.24	27.41	36.95
PCU	20.30	19.33	20.59
PCU + K-means	20.24	19.32	19.29

Table 2. Comparison of the results for the S datasets

Method	S sets			
	S1 ($*10^7$)	S2 ($*10^8$)	S3 ($*10^8$)	S4 ($*10^8$)
K-means	134.44	13.27	16.88	15.70
RLS	89.17	13.27	16.88	15.70
PAC	840.48	77.34	57.11	63.40
PAC + K-means	143.54	18.65	16.88	15.70
PCU	89.18	13.29	16.94	15.91
PCU + K-means	89.17	13.27	16.88	15.70

Table 3. Comparison of the results for the image datasets

Method	Real datasets			
	House	Bridge	Camera	Missa
K-means	36.4	365	278	9.64
RLS	35.6	364	270	9.50
PAC	51.6	430	355	13.07
PAC + K-means	39.3	366	276	10.05
PCU	37.3	377	295	9.99
PCU + K-means	36.1	365	273	9.69

The results of the principal curve clustering are significantly better than those based on principal axis. This is especially observed for the more complicated datasets A and S, where PCU performs better than PAC + K-means. The difference between the two methods reduces for the real datasets. The real datasets present high linear correlation that enables the PAC algorithm to obtain a good result.

The comparison with repeated K-means and RLS show that the results are very close to each other and to the global optimum (see the repeating values in the tables that represent the global optimum as well).

5 Conclusions

In this work we have considered solving the clustering problem using one-dimensional projections of the dataset. We have continued studying the principal axis clustering and provide a possibility to revise the sequence in the sense of minimum weight Hamiltonian path.

The main part of the work concentrates on clustering along the principal curve. The principal curve has the advantage of being a non-linear projection that can model diverse types of data. The tests have shown that the principal curve can be successfully applied in clustering for complex datasets. The method proves to perform also on multidimensional datasets.

For future, the parameterization of the principal curve should be improved by automatically optimizing the value of the penalty coefficient; the number of clusters should be considered as well.

References

1. Slagle, J.L., Chang, C.L., Heller, S.L.: A Clustering and Data-Reorganization Algorithm. IEEE Transactions on Systems, Man and Cybernetics, Vol. 5 (1975) 121-128
2. Wu, X.: Optimal Quantization by Matrix Searching. Journal of Algorithms, Vol. 12 (1991) 663-673
3. Soong, F.K., Juang, B.H.: Optimal Quantization of LSP Parameters. IEEE Transactions on Speech and Audio Processing, Vol. 1 (1993) 15-24

4. Jain, A.K., Murty, M. N., Flynn, P.J.: Data Clustering: A review. ACM Computing Surveys, Vol. 31 (1999)
5. MacQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1 (1967) 281-296
6. Zahn, C.T.: Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. IEEE Transactions on Computers (1971) 68-86
7. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall, New Jersey (1988)
8. Bezdek, J.C., Ehrlich, R., Full, W.: FCM: the Fuzzy c-Means Clustering Algorithm. Computers and Geosciences, Vol. 10 (1984) 191-203
9. Kohonen, T.: Self-Organizing Maps. Springer, Berlin Heidelberg (1995)
10. Fränti, P.: Genetic Algorithm with Deterministic Crossover for Vector Quantization. Pattern Recognition Letters, Vol. 21 (2000) 61-68
11. Gordon, A.D.: Classification. Chapman and Hall, London (1980)
12. Wu, X.: Color Quantization by Dynamic Programming and Principal Analysis. ACM Transactions on Graphics, Vol. 11 (1992) 348-372
13. Aggarwal, A., Schieber, B., Tokuyama, T.: Finding a Minimum Weight K-link Path in Graphs with Monge Property and Applications. In: Proceedings of the 9th Annual Symposium on Computational Geometry (1993) 189-197
14. Johnson, R.A., Wichern, D.W.: Applied Multivariate Statistical Analysis. Prentice-Hall, New Jersey (1988)
15. Garey, M., Johnson, D.: Computers and Intractability: A Guide to NP_Completeness. W.H. Freeman, New York (1979)
16. Hastie, T., Stuetzle, W.: Principal Curves. Journal of the American Statistical Association, Vol. 84 (1989) 502-516
17. Banfield, J.D., Raftery, A.E.: Ice Floe Identification in Satellite Images Using Mathematical Morphology and Clustering about Principal Curves. Journal of the American Statistical Association, Vol. 87 (1992) 7-16
18. Chang, K., Ghosh, J.: Principal Curves for Non-Linear Feature Extraction and Classification. In: Proceedings SPIE, (1998) 120-129
19. Kegl, B., Krzyzak, A., Linder, T., Zeger, K.: Learning and Design of Principal Curves. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22 (2000) 281-297
20. Verbeek, J.J., Vlassis, N., Krose, B.: A k-Segments Algorithm for Finding Principal Curves. Pattern Recognition Letters, Vol. 23 (2002) 1009-1017
21. Sandilya, S., Kulkarni, S.R.: Principal Curves with Bounded Turn. IEEE Transactions on Information Theory, Vol. 48 (2002) 2789-2793
22. Mulier, F., Cherkassky, V.: Self-organization as an Iterative Kernel Smoothing Process, Neural Computation, Vol. 7 (1995) 1165-1177
23. Fränti, P., Kivijäri, J.: Randomized Local Search Algorithm for the Clustering Problem. Pattern Analysis and Applications, Vol. 3 (2000) 358-369

Acknowledgments

The work is based on the Master's thesis of Ioan Cleju for his M.Sc. degree under 'International Master's Program in Information Technology (IMPIT)' supported by the University of Joensuu, Finland. The work of Xiaolin Wu was supported in part by NSERC, NSF and Nokia Research Fellowship. Ioan Cleju's work is currently supported by DFG Graduiertenkolleg/1042 'Explorative Analysis and Visualization of Large Information Spaces' at University of Konstanz, Germany.