Clustering by Principal Curve with Tree Structure

Ioan Cleju[1], Pasi Fränti[2], Xiaolin Wu[3]
[1] Multimedia Signal Processing Group, Department of Computer and Information Science, University of Konstanz, Konstanz, Germany
cleju@inf.uni-konstanz.de
[2] Speech and Image Processing Research Group, Department of Computer Science, University of Joensuu, Joensuu, Finland
franti@cs.joensuu.fi
[3] Multimedia Computing and Communications Laboratory, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, Canada
xwu@mail.ece.mcmaster.ca

***Abstract*—Data clustering is intensively used in signal processing in tasks such as multimedia compression, segmentation and pattern matching. In this work we extend the use of principal curves in clustering for complex multidimensional datasets. The use of principal curve in clustering is limited for complex data. Automatic parameterization of the principal curve to assure good results for different datasets is a difficult task. We propose to use the tree structure to capture the general settlement of the data. From this topology regions of the dataset can be extracted, individually clustered using the principal curve and then optimally combined. The experiments show the improvement of the new method over the principal curve based clustering and the good performance compared to other clustering methods.**

I. INTRODUCTION

Clustering is a general unsupervised classification method that assigns the objects of a set to different classes depending on their characteristics. Most clustering applications are related to data mining, data compression and pattern matching applications in the fields of computer science, engineering or bioinformatics. Due to the general classification purpose of clustering analysis, the applications are not limited to the mentioned fields.

The research in clustering algorithms is motivated by the importance of the problem in numerous applications and thus the necessity for good and fast algorithms. The clustering problem is known to be NP-complete [1].

The K-clustering problem is a simplification of the general clustering problem that considers the number of clusters as a parameter. This work is on K-clustering problem, although for simplicity we will refer to it as clustering problem.

Each cluster is represented by a code-vector, which is the centroid of the cluster. The set of code-vectors forms the codebook. The representation error for one point is defined as the distance from the point to the corresponding code-vector. We will consider the mean squared error (MSE) as the objective function. The total sum of squared errors defines the distortion of the codebook. The goal is to optimize the MSE, which is equivalent to minimizing the distortion.

A. Related Work

The clustering algorithms are very diverse; a good overview can be found in [2]. At the top level one can differentiate hierarchical and partitional approaches. Hierarchical algorithms produce a series of partitions, known as dendogram, while partitional ones produce one partition. The methods for dendogram generation are top-down for hierarchical divisive algorithms, and bottom-up in agglomerative clustering algorithms. The most used algorithms in cluster analysis are squared error algorithms, such as K-means [3]. The performance of K-means is highly dependent on the initialization, and therefore this method is usually used to fine-tune a solution given by other algorithms. Graph theoretic algorithms model the data as a graph (e.g. minimum spanning tree), deleting the expensive edges [4]. Mixture resolving approaches assume that the data was generated by an unknown distribution and try to determine its parameters [5]. Fuzzy algorithms [6] and artificial neural networks (self-organizing maps) [7] have been used for clustering. New approaches using genetic algorithms give also good results [8].

Scalar quantization problem is a special case of clustering that can be optimally solved in linear time [9, 10]. The reason is that the optimal clusters are formed as subsequences of the whole data sequence, as the scalar data set can be naturally ordered. Order constrained clustering is a special case of vector clustering that assumes that the data is ordered, and the clusters have to be found as subsequences. The same algorithm that finds the optimal solution for scalar quantization can be applied to order constrained clustering.

The scalar quantization and the order constrained clustering problems can be both reformulated as minimum weight K-link path problems [11]. An oriented graph is constructed for the ordered data set, having 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 shortest path consisting of K edges from the first to the last node in the sequence corresponds to the optimal clustering of the set. It can be optimally found by a dynamic programming procedure [12].

In vector space, the optimal solution for the constrained clustering is not necessarily optimal for the unconstrained

clustering problem. Relative to the unconstrained formulation, the quality of the solution obtained by optimal constrained clustering is dependent on the order relation of the data.

Different possibilities to obtain a good order for clustering have been studied in [13]. Except the basic approach that uses the principal axis projection [14], other two methods are proposed. One of them considers tuning the order in the sense of minimum weight Hamiltonian path and the other one considers using the principal curve to sequence the data.

B. Clustering Based on Principal Curve

The principal curve has been developed as a natural generalization of the principal component analysis. Among the different approaches to principal curves [15, 16, 17, 18, 19, 20], we have we have chosen in [13] to use the principal curve with length constraint [18] for clustering. The main reason for the choice is that this curve minimizes the distortion of the points to the curve. Hereafter we will refer to this curve as just the principal curve.

The clustering based on principal curve performs next steps:

- constructs the principal curve,
- projects data points on the curve and sorts them,
- finds optimal clustering for order constrained formulation,
- forms Voronoi cells, and
- (optional) fine tunes the results by K-means.

The curve and the clustering algorithm that uses it need a parameter that controls the curvature and indirectly the curve length. The results of clustering are highly dependent on this parameter. Although we have proposed in [13] a range for this parameter, it does not assure good results for all types of datasets. The algorithm might not perform well for complicated datasets that cannot be meaningfully modeled by curves. Therefore we develop a new method that combines the tree structure and the principal curve to better model and cluster datasets. In section 2 we introduce the method and in section 3 we show the experimental results. Section 4 presents the conclusions and future work.

II. PRINCIPAL CURVE WITH TREE STRUCTURE CLUSTERING

The limitations of the principal curve based clustering come from the fact that the principal curve is not a structure that can model the data distribution for complicated datasets. A way to overcome this problem is to split the data space and to apply the algorithm hierarchically. We developed a method to split the data set and create subsets that can be modeled by principal curves. We cluster each subset using the principal curve based algorithm, and then we create the codebook of the whole data set combining the partial codebooks.

A pre-clustering algorithm followed by minimum spanning tree (MST) of the codebook can offer a good model of the data (see Figure 1). The MST of the codebook is not expensive to construct, as the codebook has a reduced size compared to whole set size. The MST can be split in branches partitioning this way the dataset. The advantage of this method is that the data subset corresponding to a branch of the tree can be meaningfully modeled by a principal curve. We consider two rules to separate the branches:

- I a node of the tree has only one descendent, both of them should belong to the same branch.
- If a node has more descendents, at most only one of them should belong to the same branch with the parent node.

A straightforward algorithm based on depth-first-search trace of the tree divides the dataset in several subsets. As it can be observed in Figure 1, the subsets can easily be modeled by the principal curve.

Clustering the subsets to the same codebook size as it resulted after pre-clustering is not a good option because this would improve the solution only locally. We propose to overcome this problem by considering multiple codebook sizes for each subset and combine them to get the optimal result.

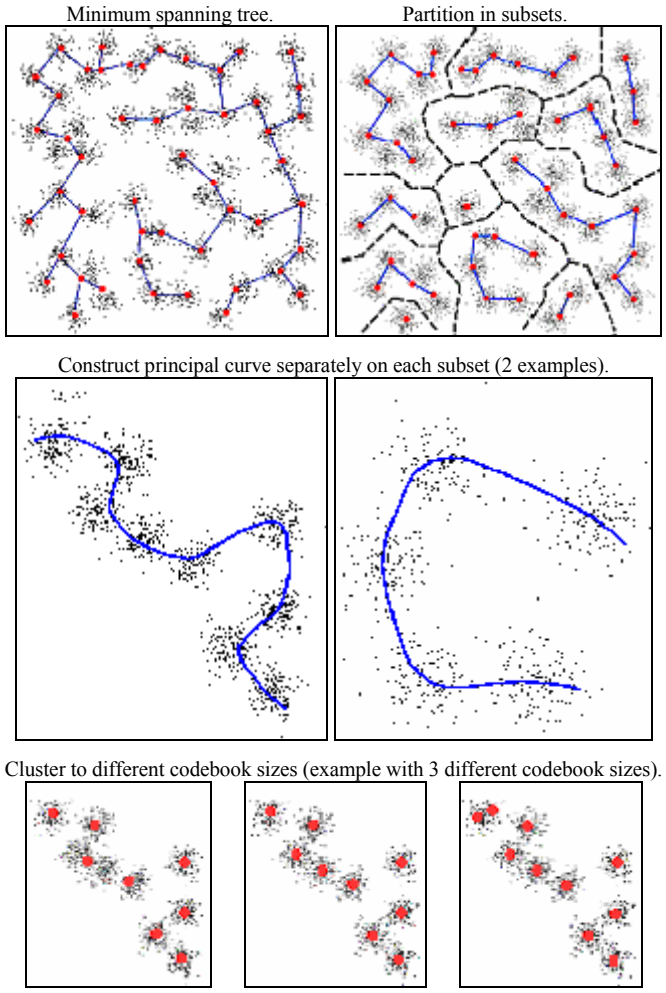

Figure 1. The main steps of principal curve with tree structure clustering.

A. Optimal Combination of Codebooks

The principal curve based clustering uses dynamic programming to cluster the data [13]. This technique constructs the solution step by step, using optimal sub-solutions. Solving

the clustering problem for the codebook size K solves clustering for all the codebook sizes smaller than K without additional computational expense.

We consider the codebook sizes after pre-clustering as estimations of the number of code-vectors in each subset. We compute codebooks with varying sizes within the neighborhood of the estimation. The optimal combination algorithm constructs the codebook of fixed size K for the whole dataset from the codebooks of subsets, minimizing the total distortion. This is a classical optimization problem that can be solved by dynamic programming [21].

B. Complexity of the Method

The complexity of the principal curve clustering algorithm is $O(K\ N^2)$ where N is the number of data points and K the codebook size. Without considering that the neighborhood range from the initial estimation of codebook sizes as negligible, the time complexity for applying the principal curve based clustering successively on all the subsets is smaller than applying it to the whole dataset. The construction of the MST and the optimal combination algorithm depend on the codebook sizes or the number of branches and these parts are usually negligible as complexity. We can conclude that without considering the pre-clustering algorithm, the complexity is upper bounded by $O(K\ N^2)$.

III. EXPERIMENTAL RESULTS

For the experiments we have used three types of data sets. The A data sets (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 also artificial two-dimensional datasets with varying complexity in terms of spatial data distributions. The real data sets (House, Bridge, Camera, Missa) come from images, representing color (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 data sets have 3 and 16 dimensions.

For the principal curve with tree structure clustering algorithm (TBC) the experiments have considered three values for the penalty coefficient: 0.01, 0.04 and 0.1. The principal curve based clustering algorithm was used for pre-clustering as well. The penalty coefficient parameter was kept unchanged during both pre-clustering and subsets clustering. For clustering based on principal curve algorithm (PCU) the MSE value was chosen as the best considering the penalty coefficient in the range 0.001 to 0.22 (15 different values). Values for codebooks tuned by K-means are also shown. The results of clustering based on principal axis also included in the table.

The comparative results include the popular K-means and randomized local search (RLS) [22]. The K-means MSE values are the best results obtained by 10 repeated trials. The MSE 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.

Results show that our improved method finds codebooks very close to the global optimum in all the studied cases. The performance of the K-means tuned algorithm is slightly better than repeated K-means. Although only 3 values of the penalty coefficient were tested for TBC, compared to 15 for PCU, the best results TBC are slightly better.

TABLE I. COMPARISON OF RESULTS FOR A DATA SETS

Method	A sets		
	A1	***A2***	***A3***
K-means	$20.24*10^5$	$19.32*10^5$	$19.29*10^5$
RLS	$20.24*10^5$	$19.32*10^5$	$19.29*10^5$
PAC	$83.00*10^5$	$156.57*10^5$	$176.59*10^5$
PAC+KM	$20.24*10^5$	$27.41*10^5$	$36.95*10^5$
PCU	$20.30*10^5$	$19.33*10^5$	$20.59*10^5$
PCU+KM	$20.24*10^5$	$19.32*10^5$	$19.29*10^5$
TBC	$20.24*10^5$	$19.42*10^5$	$19.36*10^5$
TBC+KM	$20.24*10^5$	$19.32*10^5$	$19.29*10^5$

TABLE II. COMPARISON OF RESULTS FOR S DATA SETS

Method	S sets			
	S1	***S2***	***S3***	***S4***
K-means	$134.44*10^7$	$13.27*10^8$	$16.88*10^8$	$15.70*10^8$
RLS	$89.17*10^7$	$13.27*10^8$	$16.88*10^8$	$15.70*10^8$
PAC	$840.48*10^7$	$77.34*10^8$	$57.11*10^8$	$63.40*10^8$
PAC+KM	$143.54*10^7$	$18.65*10^8$	$16.88*10^8$	$15.70*10^8$
PCU	$89.18*10^7$	$13.29*10^8$	$16.94*10^8$	$15.91*10^8$
PCU+KM	$89.17*10^7$	$13.27*10^8$	$16.88*10^8$	$15.70*10^8$
TBC	$89.17*10^7$	$13.30*10^8$	$16.90*10^8$	$15.88*10^8$
TBC+KM	$89.17*10^7$	$13.27*10^8$	$16.88*10^8$	$15.70*10^8$

TABLE III. COMPARISON OF RESULTS FOR IMAGE DATA SETS

Method	Image data sets			
	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+KM	39.3	366	276	10.05
PCU	37.3	377	295	9.99
PCU+KM	36.1	365	273	9.69
TBC	38.25	372	289	10.10
TBC+KM	36.19	365	277	9.62

IV. CONCLUSIONS

The results of the new algorithm are slightly than the basic algorithm based on principal curve. The biggest improvement of the new method consists in the fact that the main problem of the principal curve based clustering, setting the penalty coefficient, is overcome. The principal curve with tree structure clustering algorithm splits the whole dataset into subsets that can be successfully clustered using the principal curve, independent on the initial data size. The results are good as compared to other methods in clustering as well.

Future work should consider in more detail the combination of the codebooks, as there are cases when the clusters are split between different subsets. Estimation of the number of clusters based on the data model proposed should be studied as well.

ACKNOWLEDGMENT

The work was accomplished at the University of Joensuu during the M.Sc. degree studies of Ioan Cleju and was possible due to the IMPIT program developed at the mentioned university. The work of Xiaolin Wu was supported in part by NSERC, NSF and Nokia Research Fellowship.

REFERENCES

[1] J. L. Slagle, C. L. Chang and S. L. Heller, "A clustering and data-reorganization algorithm", IEEE Transactions on Systems, Man and Cybernetics, vol. 5, pp. 121-128, 1975.

[2] A.K. Jain, M. N. Murty and P.J. Flynn, "Data clustering: a review", ACM Computing Surveys, vol. 31, no. 3, September 1999.

[3] J. MacQueen, "Some methods for classification and analysis of multivariate observations", In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, vol. 1, pp. 281-296.

[4] C.T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters", IEEE Transactions on Computers, C-20, January 1971, pp. 68-86.

[5] A. K. Jain and R. C. Dubes, Algorithms for Clustering Data, Prentice-Hall, March 1988, New Jersey, US.

[6] J. C. Bezdek, R. Ehrlich and W. Full, "FCM: the fuzzy c-means clustering algorithm", Computers and Geosciences, vol. 10, 1984, pp. 191-203.

[7] T. Kohonen, "Self-organizing maps", Springer Series in Information Sciences, vol. 30, December 2000, Berlin, Germany.

[8] P. Fränti, "Genetic algorithm with deterministic crossover for vector quantization", Pattern Recognition Letters, vol. 21, no. 1, January 2000, pp. 61-68.

[9] X. Wu, "Optimal quantization by matrix searching", Journal of Algorithms, vol. 12, issue 4, December 1991, pp. 663-673.

[10] F. K. Soong and B. H. Juang, "Optimal quantization of LSP parameters", IEEE Transactions on Speech and Audio Processing, vol. 1, issue 1, January 1993, pp. 15-24.

[11] A. Aggarwal, B. Schieber and T. Tokuyama, "Finding a minimum weight k-link path in graphs with monge property and applications", In: Proceedings of the 9th Annual Symposium on Computational Geometry, San Diego, California, United States, May 1993, pp. 189-197.

[12] A. D. Gordon, Classification, Chapman and Hall, 1980, London, UK.

[13] I. Cleju, P. Fränti and X. Wu, "Clustering based on principal curve", Scandinavian Conference on Image Analysis (SCIA'05), Joensuu, Finland, June 2005 (submitted).

[14] X. Wu, "Color quantization by dynamic programming and principal analysis", ACM Transactions on Graphics, vol. 11, issue 4, October 1992, pp. 348-372.

[15] T. Hastie and W. Stuetzle, "Principal curves", Journal of the American Statistical Association, vol. 84, no. 406, June 1989, pp. 502-516.

[16] J. D. Banfield and A. E. Raftery, "Ice floe identification in satellite images using mathematical morphology and clustering about principal curves", Journal of the American Statistical Association, vol. 87, no. 417, March 1992, pp. 7-16.

[17] K. Chang and Ghosh Joydeep, "Principal curves for non-linear feature extraction and classification", In: Proceedings SPIE, vol. 3307, April 1998, pp. 120-129.

[18] B. Kegl, A. Krzyzak, T. Linder and K. Zeger, "Learning and design of principal curves", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 3, March 2000, pp. 281-297.

[19] J.J. Verbeek, N. Vlassis and B. Krose, "A k-segments algorithm for finding principal curves", Pattern Recognition Letters, vol. 23, issue 8, June 2002, pp. 1009-1017.

[20] S. Sandilya and S. R. Kulkarni, "Principal curves with bounded turn", IEEE Transactions on Information Theory, vol. 48, issue 10, October 2002, pp. 2789-2793.

[21] R. Bellman, Dynamic programming, Princeton University Press, Princeton, 1957.

[22] P. Fränti and J. Kivijäri, "Randomized local search algorithm for the clustering problem", Pattern Analysis and Applications, vol. 3, 2000, pp. 358-369.