# REFERENCE LINE APPROACH FOR VECTOR DATA COMPRESSION

*Alexander Akimov, Alexander Kolesnikov, Pasi Fränti*

Department of Computer Science, University of Joensuu
Box 111, FIN-80111, Joensuu, FINLAND
{akimov, koles, franti}@cs.joensuu.fi

## ABSTRACT

A lossy compression algorithm for vector data based on vector quantization with preliminary polygonal approximation is considered. The main idea of the proposed approach is the use of reference lines to reduce redundancy of input vector data. The references lines are constructed as coarse polygonal approximation of input curves and compressed by a lossless algorithm. The residual vectors are then encoded by vector quantization. The proposed algorithm achieves a better rate-distortion performance than the previous cluster-based algorithm. The achieved results are close to the optimized single-level DPCM modeling scheme.

## 1. INTRODUCTION

The goal of vector map compression is to find a compact representation of map, with some limited sacrifice of spatial accuracy. Lossy compression schemes are acceptable as long as the systems can accurately infer the places of interests (e.g. a video store, train route, road name and city block) with a location device such as a GPS. Cartographers routinely use generalization to highlight key features in a map by introducing bounded distortions, e.g. errors in the location of spatial objects [1]. It is also important to have several levels of map representation for different resolutions.

Lossy compression technique can be applied to vector data. The compression procedure consists of three steps: (1) transformation of input vector data according to data model to reduce redundancy of the data, (2) vector quantization of the transformed data to create a dictionary, and (3) encoding of the transformed data using the dictionary (Fig. 1).

The redundancy of the input vector data is reduced by using polygonal approximation of the curves. The set of reference points is built as a polygonal coarse approximation of the digital curves. The rest of the input data is encoded relative to the reference lines using arithmetic coding.

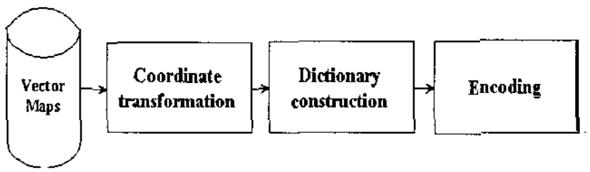


**Figure 1**. Vector quantization based compression.

## 2. LOSSY COMPRESSION OF VECTOR DATA

### 2.1 Cluster-based compression

The overall framework of encoding scheme with vector quantization can be described as follows. Vertices of digital curve $P$ are presented as a sum of predictor and error of prediction. In [1,2] the prediction is done by previous vertex [1,2], or the beginning of the curve [1]. Here and later we will denote the prediction errors as vectors $\Delta_{i,i-1}$ and $\Delta_{i,0}$ respectively, where the elements of the vector are the prediction errors of the vertex coordinates.

Lossy encoding of the transformed data is performed with quantization method. Dictionary of a given size is created for the data, the vectors $\Delta_{i,i-1}$ or $\Delta_{i,0}$ are encoded as indices to this dictionary.

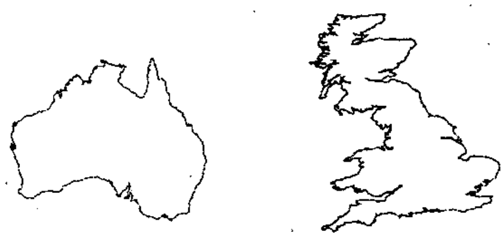

**Figure 2**. Test data sets of vector data: #1 shoreline of Australia with 2904 vertices; #2 shoreline of Britain with 10910 vertices.

**Figure 3**. An example of reference lines obtained with polygonal approximation. The points of the reference line are marked with dots.

Basically, the cluster-based compression (CBC) algorithm consists of vertices representation in $\Delta_{i,0}$ or $\Delta_{i,i-1}$ vectors with following quantization of them by some clustering algorithm. In case of $\Delta_{i,0}$ representation this algorithm will meet with very wide range of different values, that will reduce the efficiency of the quantization.

As for the case of $\Delta_{i,i-1}$ scheme, this algorithm will work efficiently for curves with small number $n$ of vertices. For the long curves, the reason of degradation of restored vector data in CBC [1] is the error propagation in restored vertices caused by lossy character of encoding relative coordinates. Dispersion $\sigma^2$ of the coordinates of restored point is proportional to the number of this point from the beginning of the curve.

The prediction scheme with closed loop [8] can be used to prevent the propagation of the error.

### 2.2 Reference line approach

Let us consider the curve with some coarse approximation (Fig. 3), which will be stored in the compressed file as low resolution representation of the vector map. Let us note a line segment between two sequential approximation points as a *reference line* [6,7]. Now consider how the information about reference lines can be used to reduce redundancy of the transformed vectors.

We perform the following affine transformation of the coordinates $(x, y)$ from the original system into a new one defined by the direction $\alpha$ of the correspondent reference line:

$$X = \cos(\alpha)(x - x_0) + \sin(\alpha)(y - y_0)$$

$$Y = -\sin(\alpha)(x - x_0) + \cos(\alpha)(y - y_0)$$

here $\alpha$ is the angle of the reference line in the coordinate system of the vector map, $(x_0, y_0)$ is the left boundary point of the reference line (Fig. 4).

In the new system of coordinates the vertical components of the vectors $\Delta_{i,i-1}$ are bounded by a given approximation error tolerance. The corresponding horizontal components are directed along the approximation line segment. With this transformation the 2-dimensional distribution became more compact, as the vectors are distributed in a narrower strip along the X-axis. The given structure allows us to use both boundary points of the as the starting points.

The proposed scheme consists in following: for prediction of each point will be used for prediction not only the previous points on curve, but also those points, which lies after this point. Basically this scheme is described by the following example (Fig. 5):

- In the beginning we know only two points: left and right boundaries of the reference line. The next predicting point will be the closest point to the left boundary (point *C*). For its prediction will be used coordinates of the known points (points *A* and *B*).
- As the point *C* is being processed and its coordinates are restored, so we will build up prediction for the closest to right boundary unprocessed point: point *D*. For creating the prediction will be used points *B* and *C*.
- If there are still unprocessed points between the boundaries of the reference line, then we repeat the algorithm, described above, repeatedly, using for prediction all point, which were processed before.

We use for prediction two points: closest processed from left side and closest processed from right side. The prediction is making in linear manner according to formulas:

$$P(x_i) = x_l \cdot q + x_r \cdot (1-q), \text{ if } i = l+1,$$

$$P(x_i) = x_r \cdot q + x_l \cdot (1-q), \text{ if } i = r-1,$$

Where $q = 1/(r-l)$. Here $l$ and $r$ are the indexes of the closest processed points from left and right side consequently. The same formulas are used for prediction of $y$ coordinate.

To prevent the propagation of error along the curve we use the closed loop approach.

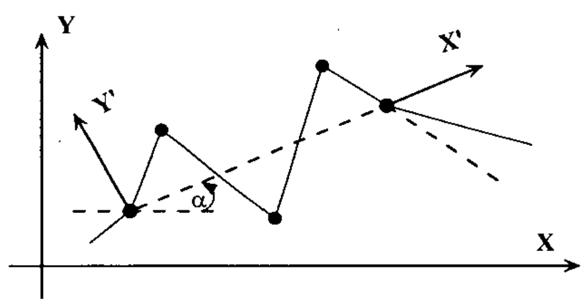


**Figure 4**. Affine transformation of segment of *P* for the correspondent reference line.

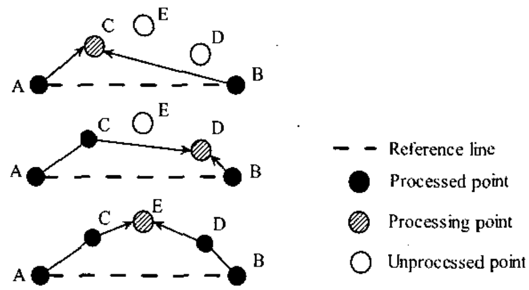

**Figure 5.** Example of prediction.

### 2.3. Dictionary construction

There are several main techniques of creating dictionaries for data encoding. The first one is to use generalized chain coding [9], multi-ring chain coding [10], or Fibonacci-Huffman-Markov (FHM) algorithm [13]. In this case a static dictionary is used, and it does not depend on the input data.

The second approach to the problem is to create a dictionary from the encoded data using clustering approach. It was observed [1] that cluster-based dictionary construction achieves lower approximation error than the fixed dictionary techniques used by FHM algorithm.

In our case we use optimal 2-D product scalar quantizer, which is built by dynamic programming algorithm [11,12]. The cost function (mean square error) is minimized by choosing partitions and number of classes at each dimension. The resulting dictionary is the product of two created dictionaries and the final centroids are pairs of elements, where both elements are elements of correspondent dictionary. They are used in the dictionary-based encoding: we assign each difference vector to the closest centroid and replace the value of the vector by the centroid's value.

## 3. EXPERIMENTAL RESULTS

Experimental series were taken for two test sets (Fig. 2). The first test data set is a long digital curve with quite smooth line; the second test data set is also a single line but with a noisy line. The points coordinates in the maps have (latitude, longitude) representation. Both maps were taken from the ESRI maps database.

There were experimental series of two types. The first series was aimed at estimating the efficiency of the proposed algorithm and compare it with the original CBC algorithm [1] and DPCM algorithm with closed loop. The second series was aimed at comparing dependency between the properties of the different reference lines and compression efficiency.

The reference lines in the test data set #1 is the approximation polygon consisting of 100 points (about 3% from total number of points). The results of the series are in Fig. 6.

Fig. 7 shows the results of compression for the data set #2, where the reference line is an approximation consisting of 100 points (about 1% from total number of points). Distortion is defined here as root mean square error (RMSE).

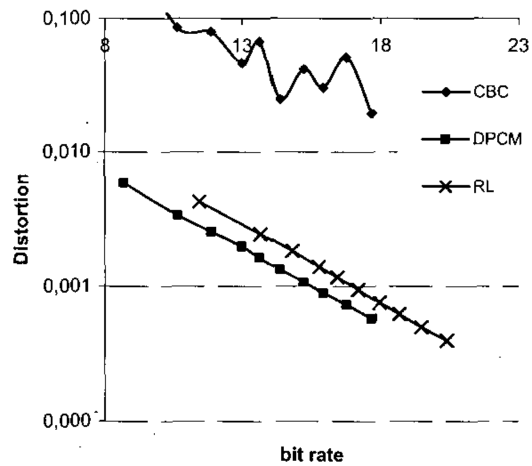

**Figure 6.** The rate-distortion curve for test set #1.

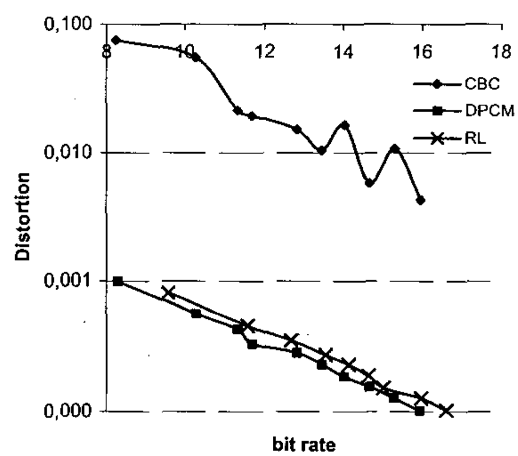

**Figure 7.** The rate-distortion curve for the test set #2.

The figures show that the proposed method slightly looses to the DPCM algorithm because of necessary to keep additional information in compressed file.

Fig. 8 illustrates the results of the second type of experiment series. We have considered the second data set with different number of points in approximation. The results in Fig. 8 shows the stability of the proposed method to the increase the number of points in coarse approximation.

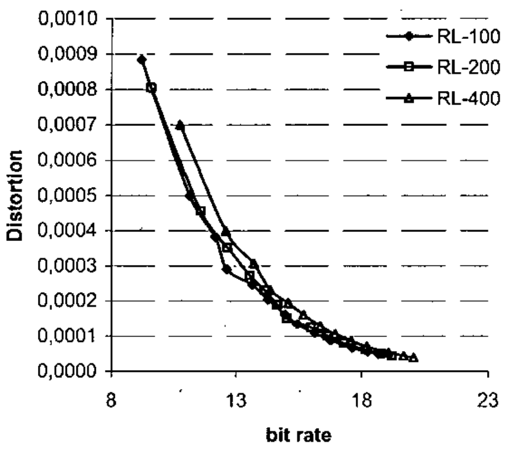


**Figure 8**. The rate-distortion curve for test set #2 for the different approximations: 100 points, 200 points and 400 points. Compression is performed with RL algorithm.

## 4. CONCLUSIONS

Lossy quantization-based compression for vector data is considered. The coarse polygonal approximation is used to reduce redundancy of the input data and create more compact. The proposed approach slightly loses to the DPCM algorithm. But instead of this the proposed method allows to store the rough representation for low-resolution mode with small looses of bit-rate. The idea can be generalized to multi-resolution approach, which is the point of future work.

## 5. REFRENCES


[1] S. Shekhar, Y. Huang, J. Djugash, C. Zhou, "Vector map compression: a clustering approach", *Proc.10th ACM Int. Symp. Advances in Geographic Information Systems-GIS'02*, pp. 74-80, 2002.

[2] K. O'Connell,"Object adaptive shape coding method", *IEEE Trans. Circuits & Syst. for Video Technol.*, vol. 7 (1) pp. 251-255, 1997.

[3] J.Dunham, "Optimum uniform piecewise linear approximation of planar curves", *IEEE Trans. Pattern Anal. Mach. Intell.* vol 8, pp. 67-75, 1986.

[4] H. Imai., M. Iri, "Polygonal approximations of a curve (formulations and algorithms)", in *Computational Morphology*, G.T.Toussaint, (Ed), pp. 71-86, North-Holland, Amsterdam, 1988.

[5] A. Kolesnikov, P. Fränti, "A fast near-optimal *min-#* polygonal approximation of digitized curves", *Proc. IASTED Int. Conf. on Automation, Control and Information Technology-ACIT'02*, pp. 418-422, 2002.

[6] A. Kolesnikov, P. Fränti, "Reduced-search dynamic programming for approximation of polygonal curves", *Pattern Recognition Letters*, vol. 24, pp. 2243-2254, 2003.

[7] A. Kolesnikov, P. Fränti, "Fast algorithm for multiple-objects *min-ε problem*", *Int. Conf. on Image Processing-ICIP'2003*, vol. 1, pp. 221-224, 2003.

[8] A.V. Oppenheim, R.W. Schafer, *Digital Signal Processing*, Prentice-Hall, 1989.

[9] H. Freeman, "On the encoding of arbitrary geometric figures", *IRE Trans. Electr. Comput.*, vol.10, pp.260-268, 1961.

[10] A.B. Johannesen, R. Prasad, N. Weyland, J. Bons, "Coding efficiency of multiring differential chain coding", *IEE Proc-I.*, vol.139 (2), pp. 224-232, 1992.

[11] J.,D., Bruce "Optimum quantization", Technical Report, MIT Research Laboratory of Electronics, 1965

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

[13] D. Solomon, *Data Compression: The Complete Reference*, $2^{nd}$ edition, Springer-Verlag, 2000.