Reprinted from Journal of the Institution of Electronics and Telecommunication Engineers, Vol. 32, No. 2, 1986 pp. 48-55

The Comparative Study of a New Pattern Recognition Based Exact Coding Method with Conventional Coding Methods for Two Tone Graphics

M.K. KUNDU AND B.B. CHAUDHURI

Electronics and Communication Sciences Unit, Indian Statistical Institute, Calcutta 700 035 India

A new Pattern Recognition (P R) based coding method for two tone contour type of graphics is introduced in this paper. The basic P R concept used here is the detection of a digital line segment in accordance with the Freeman's definition of digital line. The relative data compressibility of the new method is compared with that of different widely used representative coding methods. The relative performances of all methods are studied both on normal and noisily degraded data. The effect of gradual increase of noise level on data compressibility is studied for all the methods. The extension of the new methods is also proposed.

Indexing terms: Two tone graphics, Image coding digital line segment, Data compression, Facsimile coding, Random perturbation

THE computer storage and digital transmission of high quality graphic images are becoming increasingly important due to their exhaustive use in different application areas *e g* Space and Satellite communication [1], Photo composition [2], Facsimile transmission [3] etc. Two tone images such as business letters, documents, weather and seismic maps, line drawings, finger prints, newspaper and book pages etc cover a substantial portion of the whole pictorial information used by modern society and civilisation. The amount of such information is huge and hence efficient coding is a very important problem for digital transmission, storage and retrieval of two tone image data. The two tone image coding schemes may be grouped into two classes, namely the exact and approximate coding schemes. The exact coding schemes do not introduce any distortion during coding process. So on decoding, the original image can be reconstructed exactly. We shall restrict our discussion in this paper only to the category of exact coding.

Manuscript received 1985 January 21; revised 1985 November 26.

The exact coding methods can further be subdivided into one dimensional and two dimensional schemes. In the former case, the coding of each scanning line depends on the relations existing among the pels of the particular line and does not have any relation on the coding process of other lines. The two dimensional coding, on the other hand, makes use of horizontal and vertical correlations that exist between pels. The 'Run length coding' and its different variations belong to the 'one dimensional exact coding' scheme.

There exist two different concepts for two dimensional coding. One of them is the simultaneous coding of n lines while the other one is line by line sequential coding, such as Predictive Differential Quantizer (PDQ), Relative Address Coding (RAC), Edge difference Coding (EDIC) etc. The other one is coding only boundary points, such as Rectangular Region Coding, Block Coding and Quad-Tree Coding etc.

Huang [4] discussed three basic heuristic concepts for exact coding, namely Skipping White, Coding only Boundary Points or contour and Pattern Recognition. Of these, one of the conventional techniques of coding only boundary points is the Contour Run length Coding (CRLC). The literature of run length coding is quite rich: Morrin [5], Stern and Heinlein [6], Freeman [7], Yamada [8]. The Freeman's direction codes [9] are most popular for this purpose. Both 4-direction and 8-direction Freeman's codes are used. Recently Saghri and Freeman proposed a generalized direction coding scheme [10].

The concept of coding only boundary points is used in the new coding scheme proposed in this paper. However, the concepts of pattern recognition and digital geometry are also incorporated in this scheme. The pattern recognition concept concerns the recognition of particular class of digital straight line segments [as defined by the digital geometry] on the boundary. It is shown that the conventional CRLC is a special case of the new coding scheme called as Digital line segment coding (DLSC) [11].

In this paper, it is shown that the data compression for a contour with many digital line segments of fine angular resolution is better by the new scheme than by the conventional CRLC. Both fixed length and variable length codewords can be used for the new scheme, the later being more efficient. The details of the 'Digital line segment coding' DLSC scheme are presented in the next section.

The efficiency of the present coding scheme is compared with that of other representative exact coding schemes stated above. The schemes are discussed briefly and the experimental results and the comparative studies are presented. To test the sensitivities of these methods towards the noise, both noise-free and noisy data were taken. The noisy situation was simulated by distorting the image boundary in a random manner.

DIGITAL LINE SEGMENT CODING [DLSC]

The present scheme, called digital line segment code (DLSC) is a generalisation of CRLC. In this scheme, the digital line segments of the contour are found and coded. The concatenation of pels representing a string may be a valid digital straight line under certain conditions. The conditions stated by Freeman [7] are:

(*a*) At most two basic directions are present in the string and these differ only by unity, modulo eight.

(*b*) If there are two directions, one of them always occurs singly.

(*c*) Successive occurrences of the principal direction occurring singly are as uniformly spaced as possible.

It was pointed out by Pavlidis [12] that the condition (*c*) is somewhat fuzzy. Rosenfeld [13] defined a chord property and showed that the necessary and sufficient condition for a chain code being the chain code of a line is the chord property.

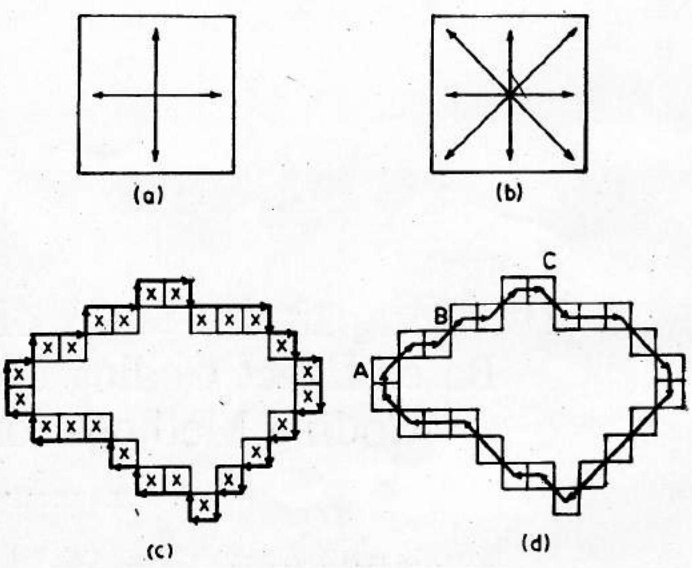


Fig 1 *a–b* Directions of 4-direction and 8-direction CRLC, respectively
c–d Representation of the same outline by 4-direction and 8-direction CRLC

It is easy to see from Fig 1*c* and Fig 1*d* that each run denotes a straight line satisfying above properties. Also, the number of runs in Fig 1*d* is less than in Fig 1*c*. This number would reduce further if some 'basic shape' of string in addition to the single pels repeating along certain direction on the contour is found out. The basic shape may be called line segment LS unit. In Fig 1*d*, it is seen that the basic shape or LS unit of the string AB is repeated thrice between A and C of the outline. Thus, the portion AC could be represented by a single run instead of six runs as in Fig 1*d* and the number of bits required to code them would become less. In fact, the portion AC is a digital straight line.

We restrict ourselves to digital line segments with slopes $\pm 1\ m+1$, $\pm m+1$ or $\pm m\ m+1$, $\pm m+1$ $m,m \in I^{+}$ with respect to the horizontal or vertical directions shown in Fig 1*b*.

Let P and Q be the two basic directions satisfying condition [*a*] of which Q occurs singly as in condition [b] given above. P or Q is one of the eight directions of Fig 1*b* satisfying the two conditions. For a positive integer *m*, let *m*PQ denote *m* successive occurrences of pels in the direction of P followed by the occurrence of a pel along Q in a string. For the kind of line segments being considered, *m*PQ is the representation of an LS unit. In fact, *m*PQ may represent digital line segments with any of the slopes $\pm 1\ m+1$, $\pm m+1$ or $\pm m\ m+1$, $\pm m+1$ *m* described above. If *m* = Q, digital lines along the eight directions of Fig 1*b* are only accounted resulting in the conventional Freeman's scheme.

As an example, consider Fig 2 where the portion ab constitutes a LS unit repeating thrice over ac. The LS unit may be represented by 3PQ where P denotes the horizontal direction and Q denotes a direction at +45° with respect to P. Similarly, the LS unit for the digital line AC of Fig 1*b* is PQ. The slope of the line in Fig 2 is $+\frac{1}{4}$ with respect to the horizontal and that of the line of portion AC of Fig 1*d* is $\frac{1}{2}$ with respect to the horizontal direction.

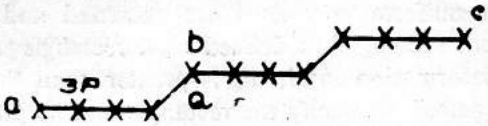
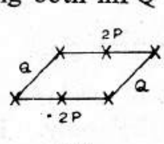


Fig 2 A discrete line segment

To code a digital outline, the following properties of a digital line are considered.

Firstly, both mPQ and QmP are the representation of LS unit of a line of the same slope, as shown in Fig 3*a*. Hence, in a digitized contour a digital straight line may begin with a singly occurring *i e*, Q type or multiply occurring P type direction. But the same codeword cannot be used for coding both mPQ or QmP because in that case the contour string cannot be reproduced exactly after decoding.



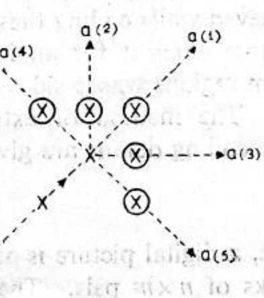


Fig 3*a* Representation of the same direction by direction units mPQ and QmP

Fig 3*b* Five possible directions of starting a new line segment

Secondly, as shown in Fig 3*b*, the angle *a*(1) between the line joining the last two pels of the previous line segment and the line joining the last pel of the pervious line segment with the first pel of the current line segment may be 0°, ±45°, ±90°. Let these angles be denoted as *a*(1), *a*(2), *a*(3), *a*(4) and *a*(5), respectively.

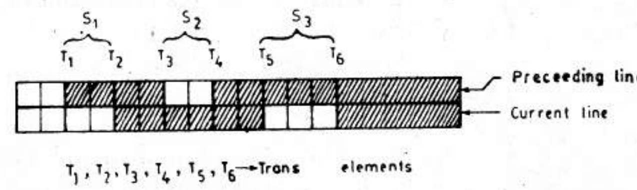


Fig 3*c* Coding strategy for EDIC scheme

Thirdly, there are two possible orientations of Q at ±45° with respect to P and vice-versa for which MPQ or QmP is a valid LS unit. The orientations may be denoted as *b*(1) and *b*(2), respectively. The detailed coding scheme is discussed in [11].

Let us define the compression ratio for DLSC as DLSC for a two-tone picture, which can be written as,

$$\delta_{\mathrm{DLSC}} = \frac{\text{Total no. of pels in the picture}}{\text{No. of bits required to represent the picture by DLSC}}$$

For a picture of size $M \times M$ with *N* different contours let there be L_i, $i=1$, *N* line segments with direction units P or *m*PQ, $m=1$, *n* present on the contours. It requires a total of $\sum_{i=1}^{N} (q+r)\, L_i$ bits to represent them. *q* and *r* represent the number of pits for LS and run respectively. Also, for each of the *N* contours, a starting pel is to be coded. It requires *K* bits to code a starting pel hence for DLSC, the compression ratio δ_{DLSC} is

$$\delta_{DLSC} = \frac{M^2}{NK + \sum_{i=1}^{N} (q+r)\, L_i} \tag{1}$$

A picture is compressible if $0 < 1/\delta_{\mathrm{DLSC}} < 1$

That of CRLC is expressed as

$$\delta_{CRLC} = \frac{M^2}{NK + \sum_{i=1}^{N} (2+r)\, L_i'} \tag{2}$$

where L_i = number of run in the i^{th} contour.

The relative compressibility δ_{rel} is given by

$$\delta_{rel} = \delta_{DLSC}/\delta_{CRLC} = \frac{NK + \sum_{i=1}^{N} (2+r)\, L_i}{NK + \sum_{i=1}^{N} (q+r)\, L_i} \tag{3}$$

A simple way of choosing the initial point is to search for a 90° bend in the pel string. If 90° bend is unavailable, a 45° bend or any other change in direction from raster may be used as starting pel. Also, at the end of each run, the strategy is to search a line segment with unit mPQ or QmP. Unit Q i e, $m=0$ is allowed only when mPQ, $m>1$ is unavailable at the current position. The details of the algorithm and the typical code book is discussed in [11].

The algorithm is used repeatedly when more than one contour is present in a picture. The number of comparisons of direction required to find a direction unit mPQ or Qmp is $m+1$. If there are Rl_i direction units in the i^{th} discrete line segment of i^{th} contour, the maximum number of comparisons C required by this algorithm

$$C = \sum_{i=1}^{N} L_i\, Rl_i\, (m+1)$$

In addition to C some comparisons are required to find out the initial point. The number of such comparisons varies from contour to contour. If the average is I_{av} then for N contours, total number of comparisons $I = N\, I_{av}$. The complexity of the algorithm is therefore of the order of $O(C)+O(I)$.

BRIEF DESCRIPTIONS OF DIFFERENT REPRESENTATIVE EXACT CODING METHODS

In this section, coding methods representing different types of one dimensional and two dimensional exact coding are briefly discussed. It is generally found that the variable length coding scheme for a particular method is superior to the fixed length coding scheme although in the former method, the costs of coding and decoding are more. In describing different methods, the variable length coding will not be discussed unless it is unavoidable for the particular method.

One-dimensional run length coding [RLC]

When an image is scanned in a raster fashion, each scan line of pel consists of sequence of grey levels. In run length encoding, this sequence of grey levels are mapped into sequence of integer pair g_k, l_k, where g_k denotes the grey levels and l_k denotes the run length. The run length is the number of adjacent picture elements having the same grey level value. In this method, scanning starts from top left most corner pel. The grey level g_1 is set equal to that of the starting pel and l_1 the length of the run. At the first grey level transition, the grey level g_k is set equal to that of the changed value and corresponding uniform grey level run as run length l_k. For two-tone case, there will only be two values of grey level namely black '0' and white '1'. The procedure continues till the complete picture is scanned. The run length may vary from 1 to N. To accommodate the largest possible run, the code length will be $> \log_2 N$ and total code length $1+\log_2 N$ where 1 bit is allocated for grey level values [3].

Contour run length coding [CRLC]

In CRLC, the image is scanned in a raster fashion, the point where the first transition occurs is selected as starting point and corresponding co-ordinate is stored.

In this scheme, generally Freeman's 4 or 8 direction chain code is used. But 8-direction chain code is more popular for encoding the different directions. The Fig 1a and 1b show the different directions for 4 and 8 direction contour coding. From the starting point, the first run begins along the direction of the next pel and continues so long as subsequent contour pels can be traversed along the same direction. Then a new run is started along a new direction. In each run, the direction and the number of pels called the run length are stored. The process continues till whole of the contour is traversed and the process is repeated for other contours. In Fig 1c and 1d, the thick solid lines with arrow marks show how the contour may be traversed for encoding. Each arrow mark on the thick line denotes the end of a run. Each code word consists of two sub code words, one for the direction and the other for the run length. If there are d possible directions and maximum run length is l then the total length of the code word will be b_d+b_l where b_d and b_l are the smallest integer greater than or equal to $\log_2 d$ and $\log_2 l$ respectively. The details of this method are described in [5].

Rectangular region coding

The rectangular region coding is one of the two-dimensional coding schemes where efficient rectangular regions of uniform grey level are extracted and coded. An 'efficient rectangle' is defined as a rectangle R_n such that the information inside R_n is greater than the information required to specify the rectangle and its grey level. If the image field is of size $2^m \times 2^n$ and the grey level of each pel be represented by k-bits, then an efficient rectangle of size $l_x \times l_y$ will be such that

$$2\,(m+n) + K < k \times l_x \times l_y$$

If a uniform region is covered by N efficient rectangles then the total number of bits saved due to the above scheme is

$$k \sum_{i=1}^{N} S_i - N(2(m+n)+k) \text{ bits}$$

where S_i is the area of i^{th} rectangle.

In general, a region of uniform grey level does not contain 'efficient rectangles' only. The regions left out after the extraction of these rectangles are coded pel by pel. No savings is achieved while coding these regions. Hence, this method is more efficient for an image having small number of uniform regions whose sides are paralled to the co-ordinate axes. The method of extraction of efficient rectangles and the coding details are given in [15].

Block coding

In this scheme, a digital picture is partitioned into two-dimensional blocks of $n \times m$ pels. The blocks are coded according to the frequency of occurrence of a type of block. Blocks with higher frequency are represented by shorter code words. However, an optimum coding for the block of size $m \times n$ will require 2^{mn} code words of varying length. To avoid such a large code book, some suboptimal schemes are used. In one of the suboptimal schemes, the most likely block configuration is coded by a single bit, represented by say, 0. The codeword for any other configuration is obtained by nm bits of the block preceded by a prefix bit say, 1.

The scheme is quite efficient in facsimile coding where a large protion ($\simeq 90\%$) of the image field is white and a large number of white blocks can be obtained through partitioning. A reasonable amount of data compression is obtained if these white blocks arc coded by single bit code word.

There are different variations of block coding schemes. An excellent review of the topic is given in [16].

Edge difference coding

This is a representative coding scheme in the class of line-by-line sequential coding. This scheme has the capability of achieving higher data compression than that of simultaneous coding of n line category. However, in this category of coding, each coded line is reconstructed utilizing the information already generated in the preceding line. If any error is introduced during the generation of any preceding line, this error is propagated in a vertical direction from top to bottom. This is a major drawback of this type of coding in comparison to the simultaneous coding of n lines. However, the error is minimized by introducing a one dimensionally coded line after every K lines. In 'Edge difference coding' EDIC scheme, the coder first generates pairs of transition elements. A pair of transition elements is classified into one of the three different states S_1, S_2 and S_3 as shown in the Fig 3c. In the state S_1 both transition elements occur on the preceding line and in the state S_3 both transition elements occur on the current line. Each pair of transition elements is coded according to its state. This method has better data compression capability than other methods in this class such as PDQ, RAC etc. The details of the method are described in [8].

RESULTS AND CONCLUSION

In the present paper, the coding performance of DLSC scheme is tested on digitized contour outlines of three different lakes of North America as shown in figures 4-6. In all the cases, a fixed length coding scheme is used, where the sub-code word for run length is such that it can accommodate the maximum run in any direction. For figures 4-6, maximum run length is found to be 9. Hence 4 bits are allotted for run. For $n=0$, two bits are used to represent the relative direction of the runs, because four relative directions at $\pm 45°$ and $\pm 90°$ are only possible for the runs.

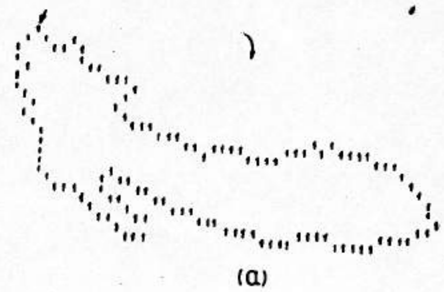


Fig 4a Digitized outline of the Lake (III) 'Huron'

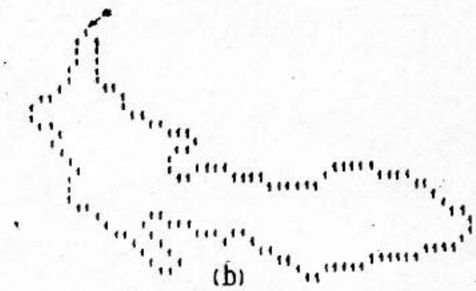


Fig 4b Fig 4a outline when noisily degraded ($\sigma^2 = 6.25$)

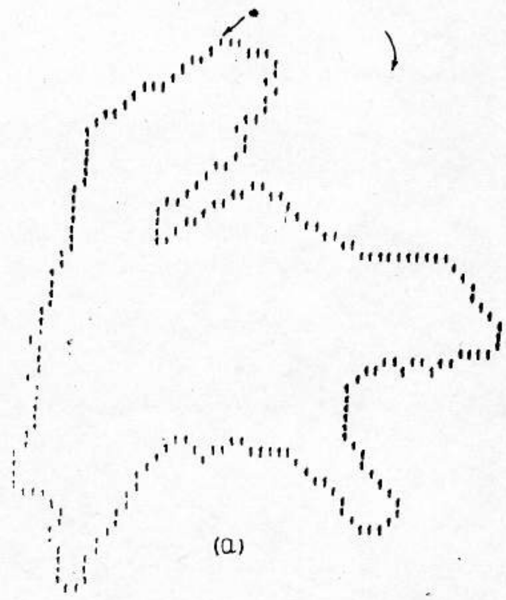


Fig 5a Digitized outline of the Lake (II) 'Michigan'

The total number of bits required to represent the contours are plotted against n, for all the the outlines as shown in Fig 7. It is seen that the minimum occurs at $n=3$ for all the digitized contours.

The efficiency of generalized scheme is tested on noisy data as well. The procedure of generating noisy outline is as discussed in [17].

Some ideal outline of different North American lakes and their noisy versions are shown in Fig 4a–6a and Fig 4b–6(b–e) respectively. The outlines and their noisy versions for different σ were subjected to CRLC and DLSC. The results are plotted in Fig 8 as δ_{rel} against σ for three lake outlines. It is seen for all the outlines that at first

Fig 5b Fig 5*a* outline when noisily degraded ($\sigma^2=6.25$)

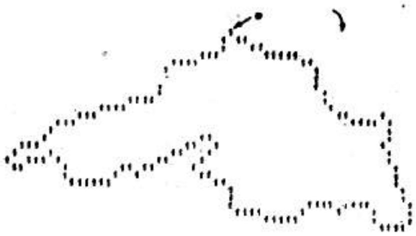


Fig 6a Fig 6*a* outline of the Lake (I) 'Superior'

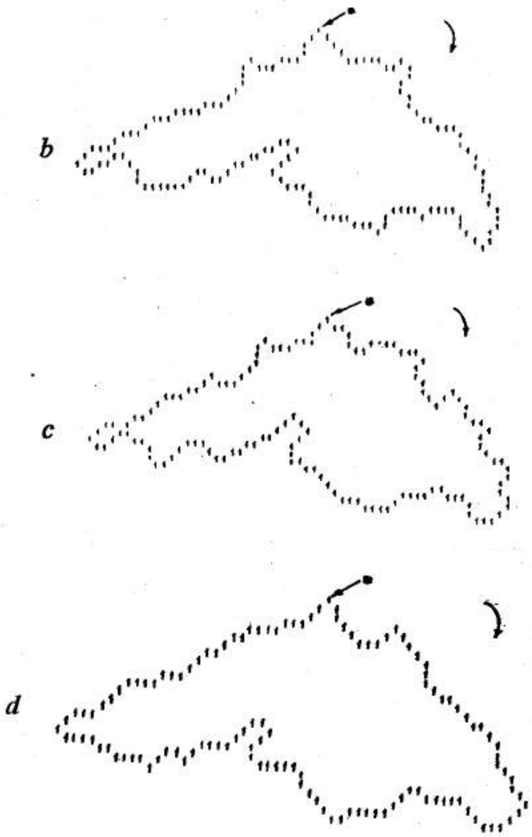


Fig 6 b–e The same outline when noisily degraded with different amount of noise σ^2 =
b 1.0; *c* 3.75; *d* 4.0; *e* 6.25

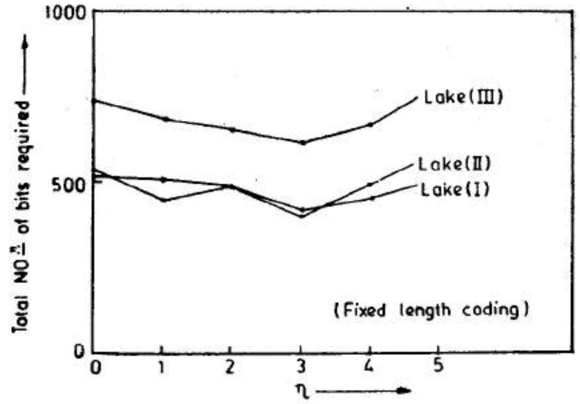


Fig 7 Bit requirement for representing Fig 4-6 with fixed length code

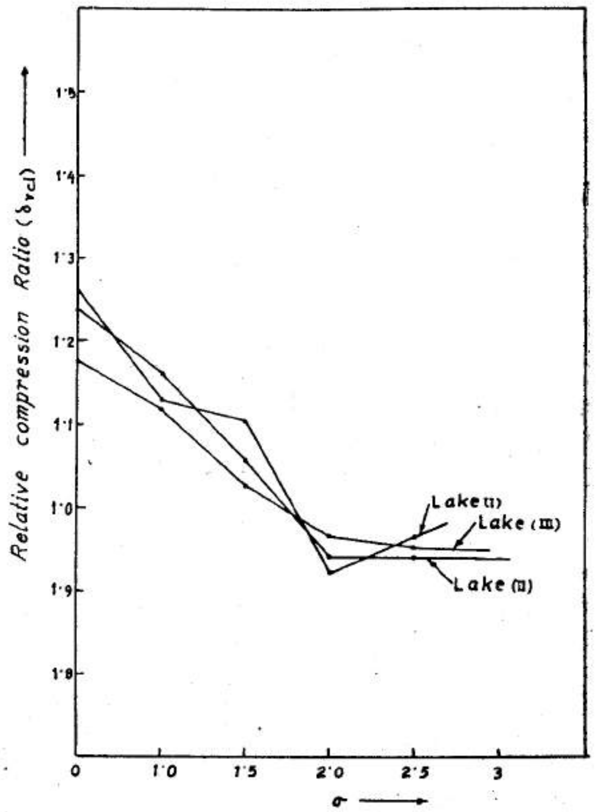


Fig 8 Variation of δ_{rel} with σ for Fig 4-6 in the case of DLSC in comparison to CRLC

δ_{rel} decreases rapidly with σ. The rate of decrease is less for moderate σ, even the rate may be negative as σ is increased further. This is apparently because excessive noise changes the shape of the outline and its line segment statistics appreciably, resulting in improvement in data compression capability.

To judge the relative merit and demerit of the DLSC scheme over diffent types of representative coding schemes existing in practice, the digitized contour of Superior lake [lake-(I)] and its noisy degraded boundaries at different noise level [Fig 6(a)-(e)] are coded by these representative coding schemes. The result is shown in Fig 9.

It is seen from Fig 9 that the total number of bits required to represent the different contours is the lowest for DLSC scheme followed respectively by CRLC, EDIC, Rectangular Region Coding, RLC and Block Coding. In case of Rectangular Region Coding and Block Coding, the compression ratio varies in both ways *i e* a decrease followed by increase as the value of σ increases.

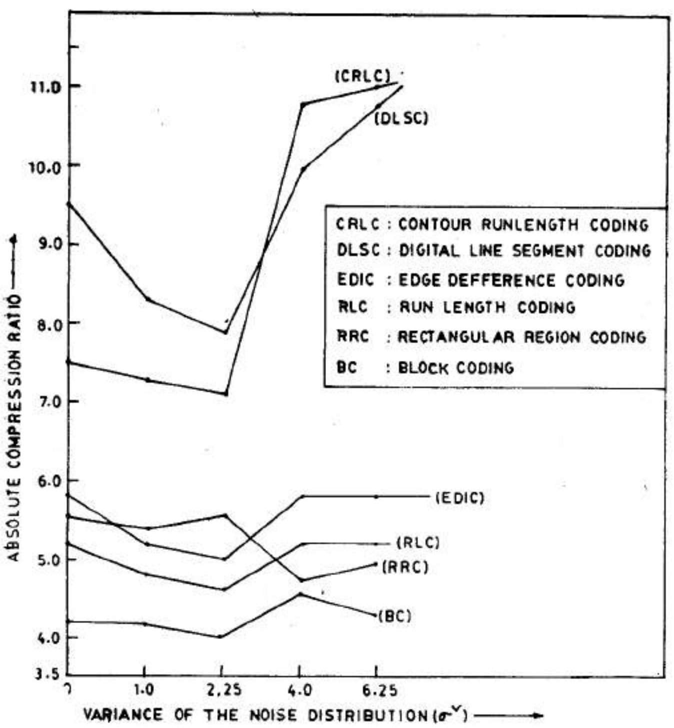


Fig 9 Variation of compression ratio (δ) with σ^2 for different representative coding schemes with 'Lake Superior' outline as test data

From Fig 9, it is clear that the compression ratios for CRLC and DLSC have similar characteristics with respect to noise. This is so because CRLC and DLSC belong to the same group of coding schemes. Similarly, RLC and EDIC show similar characteristics because of the same reason. The same inference can also be drawn for the pair of Rectangular Region Coding and Block coding schemes.

Although it is found that DLSC and CRLC are superior to all other methods discussed in this paper in terms of compression ratio for the given set of data, so it is not current to draw a generalized conclusion. It is observed that the other methods are superior when these are used for coding of silhouette type of picture data. Moreover, to code silhouette type of picture data in either CRLC or DLSC, the following additional steps will be required.

(a) Extraction of all contours from silhouette picture with the help of standard algorithms [18]

(b) During reconstruction, the contours of the picture are generated first through decoding and then the intermediate portion is filled up by any standard filling algorithm[18].

The execution of the above algorithms requires some computer time which increases the cost. However, for the contour type data or edge maps, the present scheme is superior. Also, for pattern matching and recognition, the present coding scheme may be used to advantage.

The direct extension of the DLSC scheme in three dimensional contour may be done in a tomographic fashion, *i e*, taking two-dimensional slices of the three dimensional contour. For better compression, however, it may be useful to encode digital planar surfaces. The problem is being currently studied in the present laboratory and the useful results will be communicated in future.

ACKNOWLEDGEMENT

The authors wish to thank Professor D. Dutta Majumder, Head of the Department of Electronics and Communication Sciences for his constant encouragement for the work & S. Cherian for his valuable help in making different computer programmes. The authors would would also like to thank Mrs. S. Dey Bhowmick for preparing the manuscript.

REFERENCES

1. A Habibi and B H Bostson, Potential digitization compression technique for shuttle video, *IEEE Trans*, vol COM-26, pp 1671-1682, Nov 1978.

2. A J Frank, High fidelity encoding of two level, high resolution images, *Proc IEEE Int Conf on communication*, Session 26, pp 5-10, 1973.

3. Y Yasuda, Over view of digital facsimile coding technique in Japan, *Proc IEEE*, vol 68, pp 830-845, Jul 1980.

4. T S Huang, Coding of two tone images, *IEEE Trans*, vol COM-25, pp 1406-1424, 1977.

5. T H Morrin, Chain link compression of arbitrary black and white images', Computer Graphics and Image Processing, vol 5, pp 172-189, 1976.

6. P A Stern & W E Heinlein, Facsimile coding using two dimensional runlength prediction, *Proc Zurich Seminar on digital communication*, Zurich 1974.

7. H Freeman, Boundary encoding and processing', in *Picture processing and psychopictories*, B S Lipkin and A Rosenfeld, eds, Academic Press, NY, pp 241-263, 1970.

8. T Yamada, Edge difference coding: A new efficient redundancy reduction technique for facsimile signal, *IEEE Trans*, vol COM-27, pp 1210-1217, Aug 1979.

9. H Freeman, On encoding of arbitrary geometric configuration, *IRE Trans*, vol EC-10, pp 260-268, 1961.

10. H Freeman and J A Saghri, Generalised Chain codes for planar curves, 4th Int. Conf. Pattern Recognition, Kyoto, Japan, 1978

11. B B Choudhuri & M K Kundu, Digital line segment coding : A new efficient contour coding scheme, *IEE Proc*, Pt. E, vol 131, pp 143-147, July 1984.

12. T Pavlidis, *Structural pattern recognition*, Springer-Verlag, N Y, 1977.

13. A Rosenfeld, Digital straight line segment, *IEEE Trans*, vol C-23, pp 1264-1269, 1974.

14. A Rosenfeld & A C Kak, *Digital picture processing*, Academic Press, NY, 1982

15. M Aoki, Rectangular region coding for image data compression, *Pattern Recognition*, vol 11, 1979, pp 297-312.

16. M Kunt & O Johnson, Block coding of graphics: A tutorial review, *Proc IEEE*, vol 68, pp 770-786, July 1980.

17. R L Kashyap & B J Oomen, A geometrical approach to polygon dissimilarity and classification of close boundaries, SCS-TR-18, Jan 1983, School of Computer Science, Carlton University, Ottawa Canada.

18. T Pavlidis, *Algorithm for Graphics and Image Processing*, Springer-Verlag, Berlin, 1982.

19. M K Kundu, B B Chaudhuri & D Dutta Majumder, A generalised digital contour coding scheme', *Computer Vision, Graphics and Image Processing*, vol 30, pp 269-278, 1985.

20. S Cherian, Information preserving data compression for two tone Images : A comparative study, M Tech Computer Science dissertation, Indian Statistical Institute, Calcutta, 1984.

* * * * * * * * *

ERRATA

1). On the page no.50, line no.2 of the 1st column should be read as "slopes $\pm$ 1/m+1, $\pm$ (m+1) or $\pm$ m/m+1, $\pm$(m+1)/m, $m \in I^{+}$".

2). On the page no.50, line no.13 of the 1st column should be read as "slopes $\pm$ 1/m+1, $\pm$ (m+1) or $\pm$ m/m+1, $\pm$ (m+1)/m described".

3). On the page no.50, "where $q = \log_2(4n-1)5$" should be followed by eqn.(1) in the 2nd column.

4). On the page no.54, 2nd coloumn, line no.2 should be read as "correct to draw generalized ......".