AD-A062 034 MARYLAND UNIV COLLEGE PARK COMPUTER SCIENCE CENTER F/G 20/6
METRICATION ERRORS AND CODING EFFICIENCY OF CWAIN-ENCODING SCHE--ETC(U)
SEP 78 D ROSEN, D PROFFITT AFOSR-77-3271
UNCLASSIFIED CSC-TR-696 AFOSR-TR-78-1504 NL

| OF |
AD
A062034

END
DATE
FILMED
3-79
DDC

(4) LEVEL II

COMPUTER SCIENCE TECHNICAL REPORT SERIES

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

78 12 04 . 050

④ LEVEL II

14 CSC-TR-696

AD A062034

18 19 11 12 24p.

TR-696 AFOSR-TR- 78-1504 September 1978
√ AFOSR-77-3271

15 9 Interim rept. 16 2304

6 METRICATION ERRORS AND CODING EFFICIENCY OF CHAIN-ENCODING SCHEMES FOR THE REPRESENTATION OF LINES AND EDGES,

17 A2

D./Proffitt
Biophysics Laboratory
Chelsea College
London, England

10 D./Rosen
Computer Vision Laboratory
University of Maryland

ABSTRACT

Errors associated with several boundary encoding schemes are discussed in terms of average errors encountered when using the schemes directly for measurement of the lengths of arbitrarily directed straight lines. Ways in which the measurement errors may be diminished are examined; the simplest is to make an allowance for the number of corners that appear when a line is represented by a 4-way code. The efficiencies of the various coding schemes are examined and the 4-way code is found to be the most efficient of the close-neighbors coding schemes, although efficiency can be further increased by use of a generalized code.

The support of the U.S. Air Force Office of Scientific Research under Grant AFOSR-77-3271 is gratefully acknowledged, as is the help of Ms. Kathryn Riley in preparing this paper. D.R. also acknowledges with gratitude the receipt of a travel grant from the Wellcome Trust.

78 12 04 050 403018

1. INTRODUCTION

The detection of object boundaries is fundamental to most methods of picture processing and optical pattern recognition. Encoding these boundaries as a numerical sequence for subsequent analysis inevitably results in sampling errors. The magnitude of the errors will depend on the coding scheme employed, although the significance of the errors will depend on the use to which the data are put and a general analysis of the effect of the errors may not be possible. Nevertheless, an appreciation of the errors, together with other considerations, may well determine which coding scheme is most advantageous in particular circumstances.

We are concerned here with errors made in determining the length of a straight line or edge directly from its coding sequence. There are certainly other ways of determining the length of a straight line, of which the most obvious is to determine its end-points and calculate the distance between them. However, this calculation may not be practicable in certain situations, e.g., if the line were only "nearly" straight; and it would be grossly in error for highly curved lines which are dealt with in the accompanying paper [1].

Coding schemes are conveniently described by the number of different possible direction vectors which may be taken from a point. The schemes first analyzed here are the 4-way and 8-way codes extensively discussed by Freeman [2], which are the most commonly used since they arise naturally if a line or scene is viewed against a square grid, and the 6-way code which arises if a hexagonal grid is used. The greater the number of direction vectors allowed from a point by a coding scheme, the more accurately may an arbitrary line be coded and the more accurately may its length be measured. Coding schemes allowing more than 8 directions, although sacrificing a sequence consisting only of nearest neighbor points on a square grid, are dealt with when we consider the ways of reducing the metrication errors of 4- and 8-way coding. We show that the average error in the measurement of length, which is about 27% for uncorrected 4-way coding, may be reduced to zero by a simple algorithm.

Finally we examine the efficiency with which a line may be encoded, in terms of the number of stored bits required by the coding scheme. This analysis shows that the 4-way code is the most efficient of the close-neighbor codes, quite apart from its metric corrigibility.

2. METRICATION ERRORS IN 4-WAY CODING

Let a long but finite line of length ℓ be placed in an arbitrary orientation on a square grid. Let one end of the line be the origin of co-ordinates and suppose that the line is of sufficient length that its other end can be considered, without significant error, to lie on a grid point. Let the inclination of the line to the x-axis be θ.

If the line is encoded by the 4-way scheme, it is represented by a sequence of grid points $\{s_i\}$, $0 \leq i \leq n$ (Fig. 1). (To avoid ambiguity we take the sequence $\{s_i\}$ all lying on one side of the line. The results which follow would be unaltered if the sequence were on the other side or were straddling the line.) The path from s_o to s_n along grid lines is then the representation of the arbitrary line under 4-way coding and the length of this path is what we refer to as the length obtained directly from the coding sequence. Its value is obviously n if the grid element is of unit length.

Define the relative error in the measurement of the length of an arbitrary straight line inclined at θ to the x-axis as

$$\varepsilon(\theta) = (\text{measured length} - \text{true length}) \; / \; (\text{true length}).$$

In the case of 4-way coding it is evident from Fig. 1 that n, the measure of the line OP, is equal to OS + SP = ℓ (cosθ + sinθ) so that

$$\varepsilon_4(\theta) = (\cos\theta + \sin\theta) - 1.$$

It follows that $\varepsilon(\theta) = 0$ for $\theta = 0$ or $\frac{\pi}{2}$ but has the value 0.414 for $\theta = \frac{\pi}{4}$. If all values of θ are equally probable, the average relative error, $\bar{\varepsilon}_4$, is readily obtained from the integral of $\varepsilon_4(\theta)$ over all angles. Because of symmetry, the integral can be restricted to the range $0 \leq \theta \leq \frac{\pi}{4}$ so that

$$\begin{aligned}\bar{\varepsilon}_4 &= \frac{4}{\pi}\int_0^{\frac{\pi}{4}} \varepsilon_4(\theta)\, d\theta \\ &= \frac{4}{\pi} - 1 \\ &= 0.273. \qquad (1)\end{aligned}$$

It is also of interest to consider the standard deviation, σ, associated with measurements at many angles of the length of the line. The standard deviation is obtained from the variance of the measurements defined by

$$\sigma_4^{\ 2} = \frac{4}{\pi}\int_0^{\frac{\pi}{4}} (\varepsilon_4(\theta) - \bar{\varepsilon}_4)^2\, d\theta \qquad (2)$$

$$= 1 + \frac{2}{\pi} - \frac{16}{\pi^2}$$

from which

$$\sigma_4 = 0.124.$$

Not only is there a large error associated with measurement of length from the sequence obtained by use of 4-way coding, but also there is a large standard deviation associated with a series of measurements of lines of the same length but differing orientations.

These large values arise because of the large angle between the direction vectors available in 4-way coding.

3. METRICATION ERRORS IN 6-WAY AND 8-WAY CODING

It is to be expected that as the number of direction vectors available from the coding scheme increases, the error and standard deviation of the measurement of length will decrease. Using corresponding geometrical constructions to that of Fig. 1 and the method of the previous section, it is readily shown that for

<u>6-way coding</u> we have

$$\bar{\varepsilon}_6 = \frac{2\sqrt{3}}{\pi} - 1 = 0.103$$

and $$\sigma_6^2 = \frac{2}{3} + \frac{\sqrt{3}}{\pi} - \frac{12}{\pi^2}$$

so that $$\sigma_6 = 0.046\,;$$

<u>for 8-way coding</u> we have

$$\bar{\varepsilon}_8 = \frac{2\sqrt{2}}{\pi} - 1 = -0.100$$

and $$\sigma_8^2 = \frac{1}{2} + \frac{1}{\pi} - \frac{8}{\pi^2}$$

so that $$\sigma_8 = 0.088.$$

In the case of 8-way coding it is to be noted that the diagonal step is counted as being of unit length. It is possible to count the diagonal step as being of length $\sqrt{2}$ -- the scheme can be designated 8 ($\sqrt{2}$)-way coding -- but this is a particular case, 2-sampling, of some coding schemes considered below.

With regard to 6- and 8-way coding, it can be seen that although their average relative errors and standard deviations are less than those of 4-way coding, both remain relatively large.

4. METHODS OF REDUCING THE AVERAGE METRICATION ERROR IN 4-WAY CODING

Correction of grid unit

If, for an arbitrary straight line encoded by the 4-way scheme the grid unit is taken to be $\frac{\pi}{4}$ (= 0.785) instead of its actual value of 1, the average relative standard error, $\overline{\varepsilon_4}$, is reduced to zero. However, the standard deviation of a series of measurements is reduced only by the same factor $\frac{\pi}{4}$ and so takes the value 0.098, i.e. nearly 10%. Nevertheless, although this correction method leaves the standard deviation unacceptably large for the measurement of random straight lines, it is very convenient for measuring the length of closed non-noisy curves, for which the standard deviation falls to very low values [1].

Corner counting

In a manner of speaking, the error in the measurement of length of an arbitrary line or curve by 4-way encoding comes about because the coding scheme introduces so many corners (Fig.1) and the question arises of whether allowance can be made for this. Alternatively, one could argue that to reduce both the average relative error and the standard deviation to minimal values, two parameters will be required and since the only two quantities immediately in evidence are the grid steps and the number of turns (corners) made on the grid, these must be incorporated into a correction algorithm.

Referring to Fig. 1, in the sector $0 \leq \theta \leq \frac{\pi}{4}$, the number of grid steps taken is, as before, $\ell(\cos\theta + \sin\theta)$ and the number of corners encountered is $(2\ell\sin\theta - 1)$. If ℓ is large enough, the latter can be approximated by $2\ell\sin\theta$. Now count each grid step as α and for each corner deduct a quantity β from the total measurement. Then the relative error in measuring directly the length of a line becomes

$$\varepsilon_4(\theta)\ (\alpha,\beta) = \alpha\cos\theta + (\alpha - 2\beta)\sin\theta - 1,$$

and the standard deviation is obtained as before (eq. (2)). By standard procedures, choose α and β so that the average relative error is zero and the standard deviation is a minimum. This yields

$$\alpha = \pi(1 + \sqrt{2})/8$$
$$= 0.948$$

and

$$\beta = \pi/8\sqrt{2}$$
$$= 0.278.$$

With these values, the average relative error is zero, as required, and the standard deviation has the value 0.023. Although a standard deviation of just under $2\frac{1}{2}\%$ is not negligible, it can be tolerated in many practical situations.

m-sampling

Another way in which length measurement may be improved is that which we call m-sampling: after the line is 4-way encoded, every mth point is selected and the length of the line is taken as $\sum_{j=o} d(m)_j$ where $d(m)_j$ is the geometric distance between s_{jm} and $s_{(j+1)m}$. Fig. 2 shows an example of 6-sampling.

In the m-sampling of a straight boundary, the vectors from one selected point of the sequence, s_j, to the next one, s_{j+m}, can have only a limited number of values. This is illustrated in Fig.3 for vectors in the first quadrant. If OP = OQ = m, then the ends of the vectors will fall on the line PQ. Furthermore, if OA is a typical vector, inclined at angle ϕ_i to the x-axis, then OB + BA = m; and if BA = i (where i, like m, is an integer) then OB = m - i and $\phi_i = \tan^{-1}\frac{i}{m-i}$.

The next available vector in Fig. 3 is OC, where DC = i+1 and the angle of that vector is $\phi_{i+1} = \tan^{-1} \frac{i+1}{m-\overline{i+1}}$.

A line OP, inclined at θ to the x-axis, where $\phi_i \leq \theta \leq \phi_{i+1}$ (Fig.4), would be measured by a series of vectors equivalent in Fig. 4 to OR + RP. Since $(OR\sin\phi_i + RP\sin\phi_{i+1}) = OP\sin\theta$, etc., it readily follows that in this case the error of measurement is

$$\varepsilon(\theta) = \frac{1}{m}\{(r_i(i+1) - r_{i+1}(i))\cos\theta - (r_i(m-\overline{i+1}) - r_{i+1}(m-i))\sin\theta - m\}$$

where $r_j^2 = (m - j)^2 + j^2$.

Write $I_i = \int_{\phi_i}^{\phi_{i+1}} \varepsilon(\theta)d\theta$ where $\phi_o = 0$ and $\phi_m = \frac{\pi}{2}$.

Write the average error for m-sampling as $\overline{\varepsilon}(m)$.

Then $\overline{\varepsilon}(m) = \frac{2}{\pi}(I_o + I_1 + \ldots + I_{m-1})$.

From this it follows, on evaluating the integrals, that

$$\overline{\varepsilon}(m) = \frac{2}{\pi}\{2 - \frac{1}{m}\sum_{i=o}^{m-1} (r_{i+1} - r_i)^2 - \frac{\pi}{2}\}$$

$$= \frac{4}{\pi}\{1 - \frac{1}{2m}\sum_{i=o}^{m-1} \{((m - \overline{i+1})^2 + (i+1)^2)^{\frac{1}{2}} - ((m-i)^2 + i^2)^{\frac{1}{2}}\}^2\} - 1 \qquad (3)$$

<u>Value of the average relative error in measurements of an m-sampled line</u>

Write $\overline{\varepsilon}(m) = \frac{4}{\pi}E_m - 1$ where E_m is obtained from eq.(3). Table 1 gives values of $\overline{\varepsilon}(m)$ and numerical forms of E_m for various values of m. It is interesting to note that, $\overline{\varepsilon}(m)$ being always positive, the average length measured increases as the sampling interval falls in such a way as to provide a straight line in a log.log plot of $\overline{\varepsilon}(m)$ vs. m (Fig.5). This closely parallels the manner in which, as discussed by Mandelbrot [3], the measured length of a coastline increases as the measuring rod is reduced.

As expected, $\overline{\varepsilon}(m)$ tends to zero as m becomes very large. This can be proved formally by referring to ΔOCA in Fig. 3 and noting that $OA = r_i$, $OC = r_{i+1}$ and $CA = \sqrt{2}$, from which it follows that

$$r_1^2 + r_{i+1}^2 - 2r_i r_{i+1} \cos(\phi_{i+1} - \phi_1) = 2$$

or

$$(r_{i+1} - r_i)^2 = 2 + 2r_i r_{i+1}(\cos(\phi_{i+1} - \phi_i) - 1).$$

As m becomes very large, $(\phi_{i+1} - \phi_i)$ becomes very small. In these circumstances, write $\cos(\phi_{i+1} - \phi_i) - 1 = -\dfrac{(\phi_{i+1} - \phi_i)^2}{2}$

$$= -\frac{m}{2r_i r_{i+1}} (\phi_{i+1} - \phi_i)$$

since for very small values, $(\phi_{i+1} - \phi_i) \approx \sin(\phi_{i+1} - \phi_i) = \dfrac{m}{r_i r_{i+1}}$.

Then it follows that

$$\operatorname*{Lt}_{m\to\infty} \frac{1}{2m} \sum_{i=o}^{m-1} (r_{i+1} - r_i)^2 = 1 - \frac{\pi}{4}$$

from which, immediately, $\operatorname*{Lt}_{m\to\infty} E_m = \frac{\pi}{4}$ and $\operatorname*{Lt}_{m\to\infty} \overline{\varepsilon}(m) = 0$.

In practice, the limiting situation would be finding the two ends of a straight line and calculating the distance between them.

It is worth remarking that 2-sampling is a version of 8-way coding in which the diagonal step has the value $\sqrt{2}$. This case has been considered by Kulpa [4] with the same result as is derived here from the general formula.

5. EXPERIMENTAL VERIFICATION

The validity of several of the formulae given in the preceding sections was confirmed by computer program. A straight line of known length but with random orientation was digitized. For each method of measurement the line (usually of length 200) was measured at fifty random orientations with the results given in Table 2. The small differences between the theoretical and experimental values can be ascribed to truncation errors.

6. RELATIVE FREQUENCY OF DIAGONAL STEPS IN ENCODED STRAIGHT LINES

Freeman [5] (see also Groen and Verbeek [6]) considers the relative frequency of unit and diagonal steps in the encoded sequence of an arbitrary straight line. He reaches his result with the assumption that lines emanate from each encoded point with uniformly random orientation. This assumption can be avoided by comparing the average relative errors obtained in the measurement of straight lines encoded by the 8-way and the $8(\sqrt{2})$-way schemes. The difference between these average errors is due to diagonal steps being counted as of length 1 or $\sqrt{2}$. Suppose that in the encoded "average line", of true length ℓ, there are n steps of which a fraction ρ are diagonal. Then, with 8-way coding, using the formula for average relative error

$$\bar{\varepsilon}_8 = \frac{n - \ell}{\ell} = \frac{2\sqrt{2}}{\pi} - 1$$

and with $8(\sqrt{2})$-way coding

$$\bar{\varepsilon}_{8(\sqrt{2})} = \frac{n(1 - \rho) + n\rho\sqrt{2} - \ell}{\ell} = \frac{8(\sqrt{2} - 1)}{\pi} - 1$$

Eliminating n/ℓ yields $\rho = \sqrt{2}-1$ which is the result of Freeman [5] and of Groen and Verbeek [6].

7. CODING EFFICIENCY

Although, in particular circumstances, one definite coding scheme may be more convenient than others, there appears to be no general rule by which the merits of different schemes may be compared. It is nevertheless possible to approach such a comparison on the basis of the results of the previous sections.

To begin with, consider the number of bits, N, which must be stored in the encoding of a straight line under the coding scheme chosen, with the implication that the smaller the number of bits stored, the higher is the coding efficiency. For a w-way code with steps of equal length, a straight line of length ℓ will, on average, be encoded by a sequence of $\ell(1 + \bar{\varepsilon}_w)$ direction vectors, and each vector will require $\log_2 w$ bits to specify it. Hence the number of bits required to encode the line is, on average, $\bar{N}_w = \ell(1 + \bar{\varepsilon}_w)\log_2 w$. Values of $\bar{N}_w/\ell$ are given in Table 3 for w = 4, 6 and 8. The table shows that 4-way coding requires, on average, significantly fewer bits to be stored for the encoding of a straight line than do either of the other two schemes.

A reduction in the amount of data stored can be brought about if the 4-way encoded line or edge is m-sampled. For a straight line of length ℓ the 4-way encoding results, on average, in $\ell(1 + \bar{\varepsilon}_4)$ direction vectors. If this sequence is m-sampled (which provides 4m direction vectors) the number of direction vectors is reduced to $\ell(1 + \bar{\varepsilon}_4)/m$ but the number of bits required to specify each one is $\log_2 4m$. Hence the number of bits required to encode the m-sampled line is, on average,

$$\bar{N}_m = \ell(1 + \bar{\varepsilon}_4)\log_2(4m)/m.$$

Values of $\bar{N}_m/\ell$ are given in Table 3. It should be noted that these values are for an m-sampled 4-way-encoded straight line; for lines which initially were encoded by the 6-way or 8-way schemes, the values of $\bar{N}_m/\ell$ would be, respectively 13.4% and 29.3% smaller.

It can be seen from Table 3 that m-sampled data of an encoded straight line can be stored efficiently, even when allowance is made for the practical consideration that bits can be stored only as integers.

The m-sampling of an encoded straight line is, of course, a special case with a limited number, $4m$, of resulting direction vectors. For an arbitrarily curved line for which the original close-neighbor coding was m-sampled, more directions would appear for each value of m, resulting in a code which is a variant of the generalized codes discussed by Freeman [7]. The form of such a code is shown in Fig. 6(a) and, in Freeman's notation, specifying an m-sampled arbitrary curve would require a $(a,\ldots,m-4,m-2,m)$-code where $a = 1$ or 2 according as m is odd or even. For this form of generalized code the number of distinct direction vectors is $(m + 1)^2$ if m is odd or $(m + 1)^2-1$ if m is even.

In the accompanying paper [1] we show that an arbitrary closed curve of length ℓ is 4-way encoded by $\ell(1 + \bar{\varepsilon}_4)$ direction vectors. Accepting this result, it follows that if the coding were m-sampled, the number of direction vectors would be $N_m = (1 + \bar{\varepsilon}_4)\log_2(m + 1)^2/m$ if m is odd or the corresponding quantity if m is even. Table 3 includes values of N_m/ℓ and the advantage of m-sampling in terms of data storage requirements is readily seen. However, this mechanical method of reducing the quantity of stored data risks losing significant features of the encoded curve. In Fig. 6(b) if A were a sampling point and with $m \geq 6$, the protrusion to the right of A would disappear. To avoid such loss of features it would be necessary to allow m to vary and the value of N_m/ℓ would increase accordingly.

Data storage requirements are not the only considerations which apply in choosing a scheme to encode a line or edge. Two other obvious properties of lines are length and shape. We have shown above that a simple algorithm (a general correction factor together with an allowance for corners) may be used to reduce on average to zero the measurement error in determining the length of a straight line directly from its 4-way encoding. This algorithm, moreover, is applicable in determining the lengths of noise-free curved lines [1]. It thus appears that considerations of metrication hardly affect the quantification of the efficiency of 4-way coding. It should be mentioned that the algorithm used with length measurement of 4-way encoded lines could be applied, with different values of the correction factors, to other coding schemes. The average relative error could not, of course, be further reduced, but the standard deviation of a series of measurements could be.

The shape of a line, even in a local region, is an undefined or imprecisely defined property unless the line is straight or is of a simple geometric shape (circular, parabolic, etc.). If, at least in a local region, the line is continuous and differentiable, its curvature at a point may be obtained. For a digitized encoded line, a form of discrete curvature may be specified at each point from the two direction vectors involving the point [8, 9], or sections of the line extending over many points may be approximated by circular arcs [10]. However, extracting such features of a line [11 - 13] involves procedures which average (one or more times, cf. [9]) over several points of the encoded sequence; and the more points included in the averaging the smaller the dependence on the coding scheme. In these circumstances, and even though intuition suggests that shape is best encoded by the available coding scheme with most direction vectors, no measure of efficiency is available nor does practical experience suggest that, say, 4-way coding is worse than 8-way coding.

8. CONCLUSIONS

If a straight line or edge is encoded as a sequence of points, the simplest way of measuring the length of the line is to count the number of steps, each of known length, through the sequence. The smaller the number of direction vectors provided by the coding scheme, the greater is the average error in determining length by this means (Fig.5); and with 4-way coding the average error is just over 27% and the standard deviation of a series of measurements is just over 12%. However, a simple algorithm reduces the average error to zero and the standard deviation to less than 2.5%. It is necessary to go to a 20-way code (5-sampling of 4-way encoding) before the average error of length measurement falls below 1%, although the standard deviation is down to 2.5% with $8(\sqrt{2})$-way coding (2-sampling of 4-way encoding) (Table 2).

In terms of data storage, 4-way encoding requires significantly less capacity, on average, than the other close-neighbor coding schemes. Sampling the 4-way encoded data at regular intervals for either a straight line or an arbitrary closed curve greatly reduces the necessary storage capacity (Table 3) but for an arbitrary curved line this advantage is offset by the risk of losing significant shape features. To avoid such loss it would be necessary to sample at irregular intervals and this would cause an increase in the number of direction vectors and the number of bits stored. Quite generally, details of shape are best preserved by coding with short direction vectors and, in the limit, by using close-neighbor coding schemes.

We conclude that, of the close-neighbor schemes, 4-way coding is the most efficient. However, more generalized coding schemes, apart from reducing the number of bits to be stored, have the advantage pointed out by Freeman and Saghri [7] of involving less processing time in the further analysis which might be undertaken of the line or boundary.

REFERENCES

1. T. J. Ellis, D. Proffit, D. Rosen and W. Rutkowski, Measurement of the lengths of digitized curved lines, University of Maryland, Computer Science Center, TR-697, September 1978.

2. H. Freeman, Computer processing of line-drawing images, Computer Surveys, 6, 1974, 57 - 97.

3. B. Mandelbrot, Fractals, Freeman & Co., San Francisco, 1977

4. Z. Kulpa, Area and perimeter measurement of blobs in discrete binary pictures, Computer Graphics and Image Processing, 6, 1977, 434 - 451.

5. H. Freeman, On the encoding of arbitrary geometric configurations, IRE Trans. Electronic Comp. (IEEE Trans. Comp.), EC-10, 1961, 260 - 268.

6. F. C. A. Groen and P. W. Verbeek, Freeman-code probabilities of object boundary quantized contours, Computer Graphics and Image Processing, 7, 1978, 391 - 402.

7. H. Freeman and A. Saghri, Generalized chain codes for planar curves, Paper delivered at the 4th International Pattern *Recognition Conference*, *Kyoto*, *November* 1978.

8. H. Freeman, On the digital computer classification of geometric line patterns, Proc. Natl Electronics Conf., 18, 1962, 312 - 324.

9. M. J. Eccles, M. P. C. McQueen and D. Rosen, Analysis of digitized boundaries of planar objects, Pattern Recognition, 9, 1977, 31 - 42.

10. B. Shapiro, Ph.D. thesis, Computer Science Center, University of Maryland, July 1978.

11. A. Rosenfeld and E. Johnston, Angle detection in digitized curves, IEEE Trans.Comp., C-22, 1973, 875 - 879.

12. W. Rutkowski and A. Rosenfeld, A comparison of corner-detection techniques for chain-coded curves, University of Maryland, Computer Science Center, TR-623, January 1978.

13. H. Freeman and L. S. Davis, A corner-finding algorithm for chain-coded curves, IEEE Trans.Comp., C-26, 1977, 297 - 303.

Table 1. Exact formulae and average errors in m-sampling of straight lines

m	No. of direction vectors	E_m	$\bar{\varepsilon}(m)$
1	4	1	0.27324
2	8	$1 - (2 - \sqrt{2})^2/2$	0.05479
		$= 2\sqrt{2} - 2$	
3	12	$1 - (3 - \sqrt{5})^2/3$	0.02556
		$= 2\sqrt{5} - 11/3$	
4	16	$1 - \{(4 - \sqrt{10})^2 + (\sqrt{10} - \sqrt{8})^2\}/4$	0.01438
		$= 2\sqrt{10} - 10 + 2\sqrt{5}$	
5	20	$1 - \{(5 - \sqrt{17})^2 + (\sqrt{17} - \sqrt{13})^2\}/5$	0.00922
		$= 2\sqrt{17} - \frac{67}{5} + \frac{2\sqrt{221}}{5}$	
6	24	$1 - \{(6 - \sqrt{26})^2 + (\sqrt{26} - \sqrt{20})^2 + (\sqrt{20} - \sqrt{18})^2\}/6$	0.00641
		$= 2\sqrt{26} - \frac{70}{3} + \frac{2\sqrt{130}}{3} + 2\sqrt{10}$	
7	28	$1 - \{(7 - \sqrt{37})^2 + (\sqrt{37} - \sqrt{29})^2 + (\sqrt{29} - 5)^2\}/7$	0.00471
		$= 2\sqrt{37} - \frac{199}{7} + \frac{2\sqrt{1073}}{7} + \frac{10\sqrt{29}}{7}$	
10	40		0.00231
30	120		0.00026
100	400		0.00002

Table 2. Experimental verification of error formulae

Coding scheme	Theoretical values $\bar{\varepsilon}$	σ	Experimental values av.error	s.d.
	Close neighbor codes			
4-way	0.273	0.124	0.277	0.120
6-way	0.103	0.046	-	-
8-way	- 0.100	0.088	- 0.081	0.077
8($\sqrt{2}$)-way	0.055	0.025	0.053	0.027
	m-sampled data			
2-sampled	0.055		0.053	0.027
3-sampled	0.026		0.025	0.015
4-sampled	0.014		0.014	0.0087
5-sampled	0.0092		0.0089	0.0057
10-sampled	0.0023		0.0021	0.0014
20-sampled	0.00058		0.00053	0.00036
	corner-corrected 4-way encoding			
	0	0.023	0.00088	0.024

Table 3. Number of bits per unit length required to encode a line

	STRAIGHT LINES		ARBITRARY CLOSED CURVES	
Coding scheme	Theoretical	Practical	Theoretical	Practical
		Close-neighbor codes		
4-way	2.546	2.546		
6-way	2.850	3.308		
8-way	2.701	2.701		
		m-sampled 4-way-encoded data		
2-sampled	1.910	1.910	1.910	1.910
3-sampled	1.522	1.698	1.698	1.698
4-sampled	1.273	1.273	1.459	1.592
5-sampled	1.101	1.273	1.317	1.528
6-sampled	0.973	1.061	1.185	1.273
7-sampled	0.874	0.909	1.091	1.091
10-sampled	0.678	0.764	0.879	0.891
20-sampled	0.402	0.446	0.559	0.573

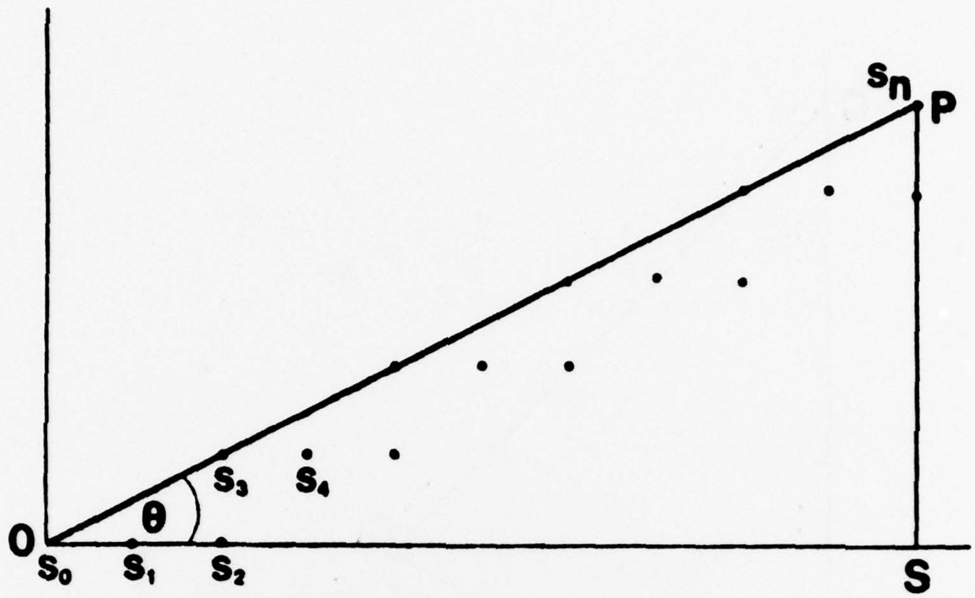

Fig. 1.

Fig. 2.

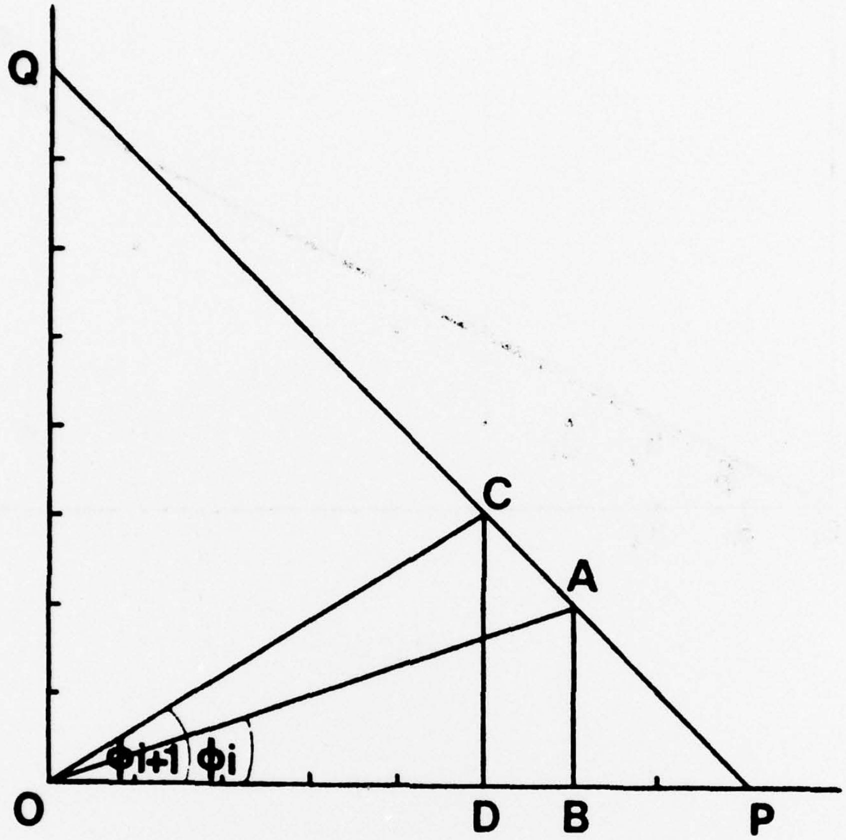

Fig. 3.

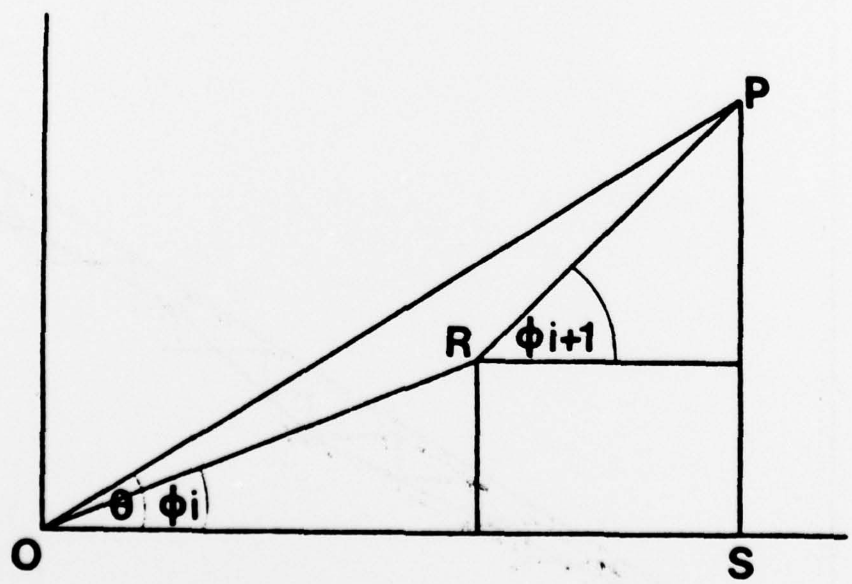

Fig. 4.

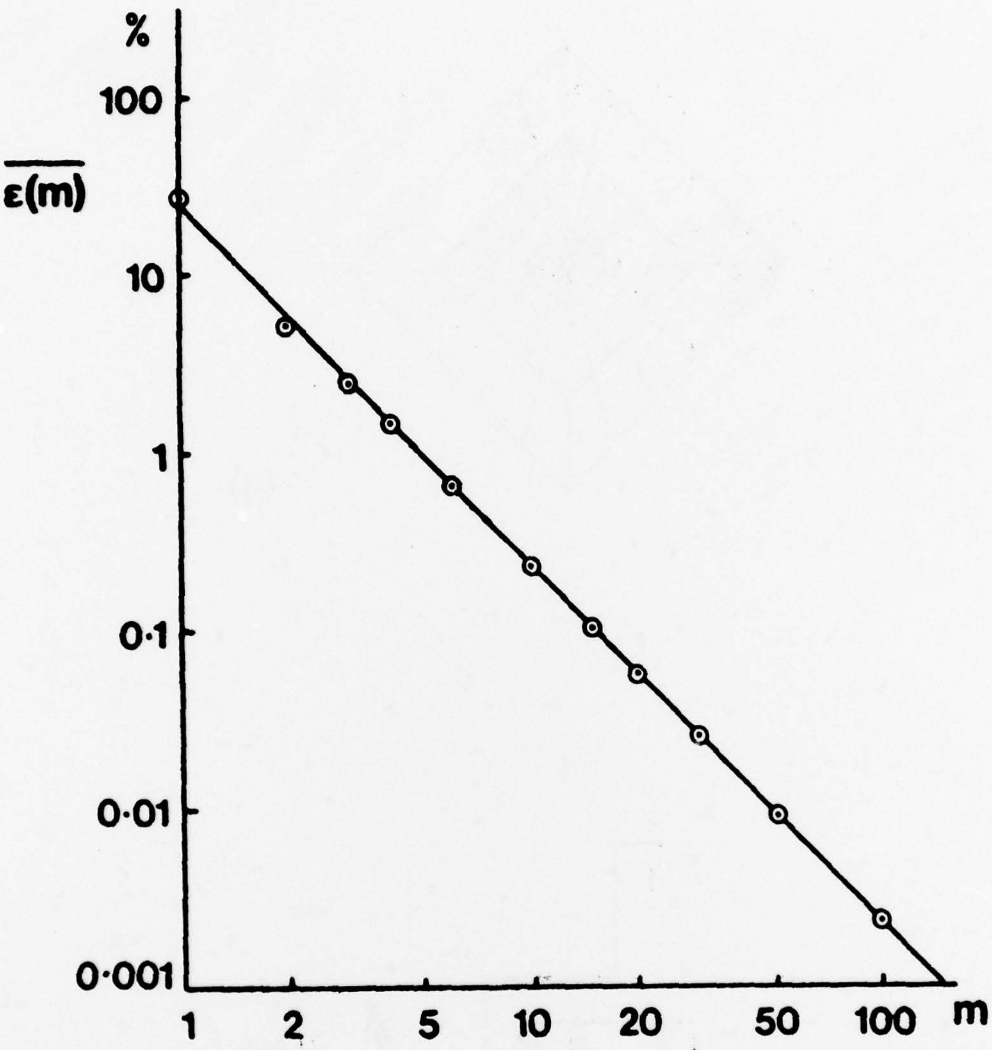

Fig. 5.

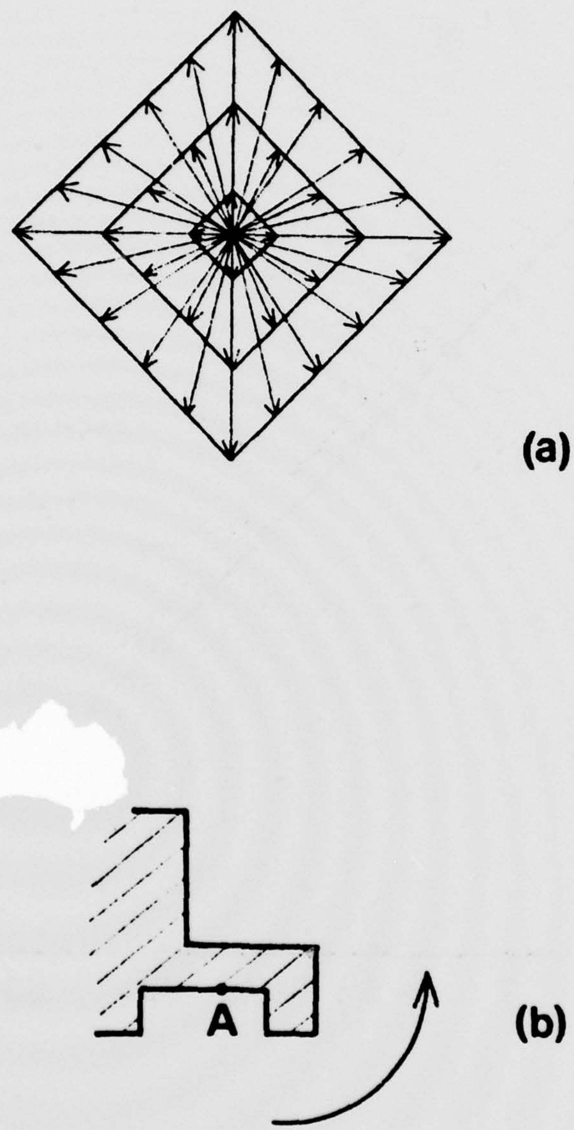

Fig. 6.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER **AFOSR-TR- 78-1504**	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) METRICATION ERRORS AND CODING EFFICIENCY OF CHAIN-ENCODING SCHEMES FOR THE REPRESENTATION OF LINES AND EDGES		5. TYPE OF REPORT & PERIOD COVERED Interim 6. PERFORMING ORG. REPORT NUMBER TR-696
7. AUTHOR(s) D. Rosen and D. Proffitt		8. CONTRACT OR GRANT NUMBER(s) AFOSR 77-3271
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Maryland Computer Science Center College Park, Maryland 20742		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM Bolling AFB, Washington, DC 20332		12. REPORT DATE September 1978 13. NUMBER OF PAGES 23
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Image processing
Pattern recognition
Chain coding
Arc length

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Errors associated with several boundary encoding schemes are discussed in terms of average errors encountered when using the schemes directly for measurement of the lengths of arbitrarily directed straight lines. Ways in which the measurement errors may be diminished are examined; the simplest is to make an allowance for the number of corners that appear when a line is represented by a 4-way code. The efficiencies of the...

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract continued.

various coding schemes are examined and the 4-way code is found to be the most efficient of the close-neighbors coding schemes, although efficiency can be further increased by use of a generalized code.