CHAPTER 9

PROCESSING LINE DRAWINGS

Godfried Toussaint

ABSTRACT

This chapter introduces the basic ideas behind Freeman coding of curves for the efficient representation and processing of line drawings. Along the way Bertrand's paradox in probability theory is illustrated with an example that utilizes random chords on a circle. Finally, using a suitable probabilistic model several chain-coding schemes are analyzed from the point of view of the probability of obtaining diagonal elements in the encoding.

1. Introduction

By a line drawing we mean an image consisting of line segments and/or curve segments which need not be connected and whose thickness carries no information. A vast quantity of images in computer graphics, image processing and pattern recognition fall under this domain. Examples of line drawings abound in engineering drawings as well as maps in automated cartography.

The problem of the efficient transmission and storage of line drawings has received considerable attention. In geographic information systems large amounts of data such as contours of maps must be stored [1]. For a recent survey of this field, the reader is referred to the article by Freeman [2]. Several methods referred to as *chain-coding* have been proposed by Freeman and his co-workers [3] - [4]. All these methods require a device to quantize the two-dimensional figure and encode it into a form suitable for machine processing and transmission. Three quantization methods of interest here will be described and compared.

Let a uniform mesh be superimposed on the line drawing. Using Freeman's notation, an x-y coordinate system can be drawn on the mesh so that every node is identifiable in terms of its coordinates (mT, nT), where m and n are integers and T represents the separation between adjacent mesh nodes. An efficient quantized description of the line drawing can then be specified by giving the coordinates of the mesh nodes which lie "close" in some sense to the given line drawing. These nodes are called "*curve points*." Different quantization schemes yield a possibly different set of curve points for a given line drawing.

Three quantization schemes frequently used in the past, referred to as *square*, *circular*, and *grid-intersect* quantization, are illustrated in Fig. 1. Consider first *square* quantization. In this case a square of side T is centered at each mesh node, with its sides parallel to the mesh lines. A node is then considered to be a curve point if any part of the line drawing falls within the square associated with that particular node. This quantization scheme applied to curve in Fig. 1 yields *abcdef* as a sequence of curve points representing the line drawing. For *circular* quantization, a circle of radius $T/2$ is centered at each mesh node. A node is then considered to be a curve point if any part of the line drawing falls within the circle associated with that particular node. This quantization

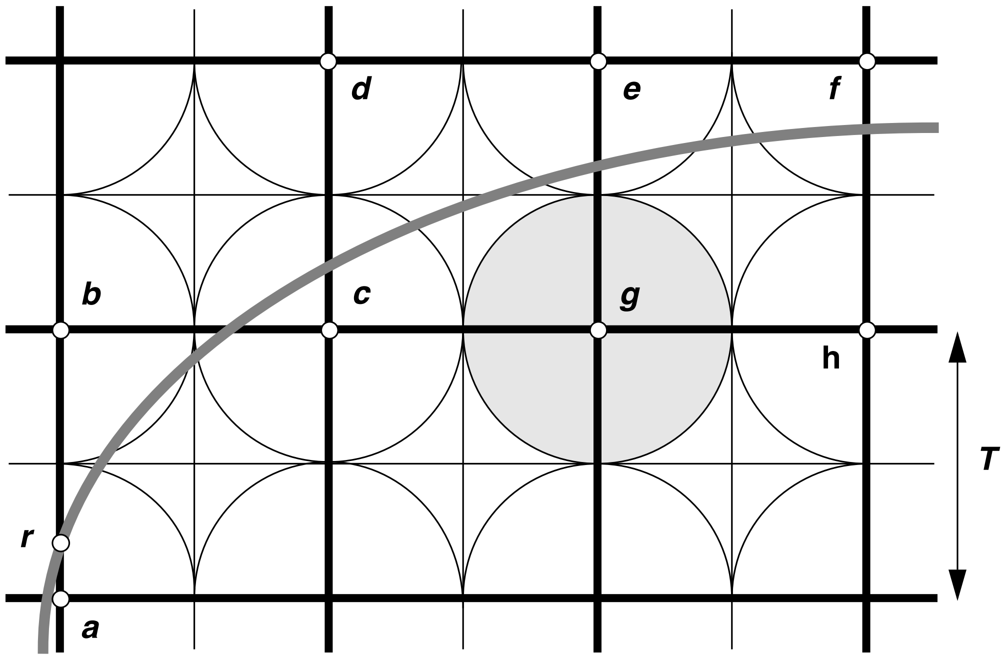
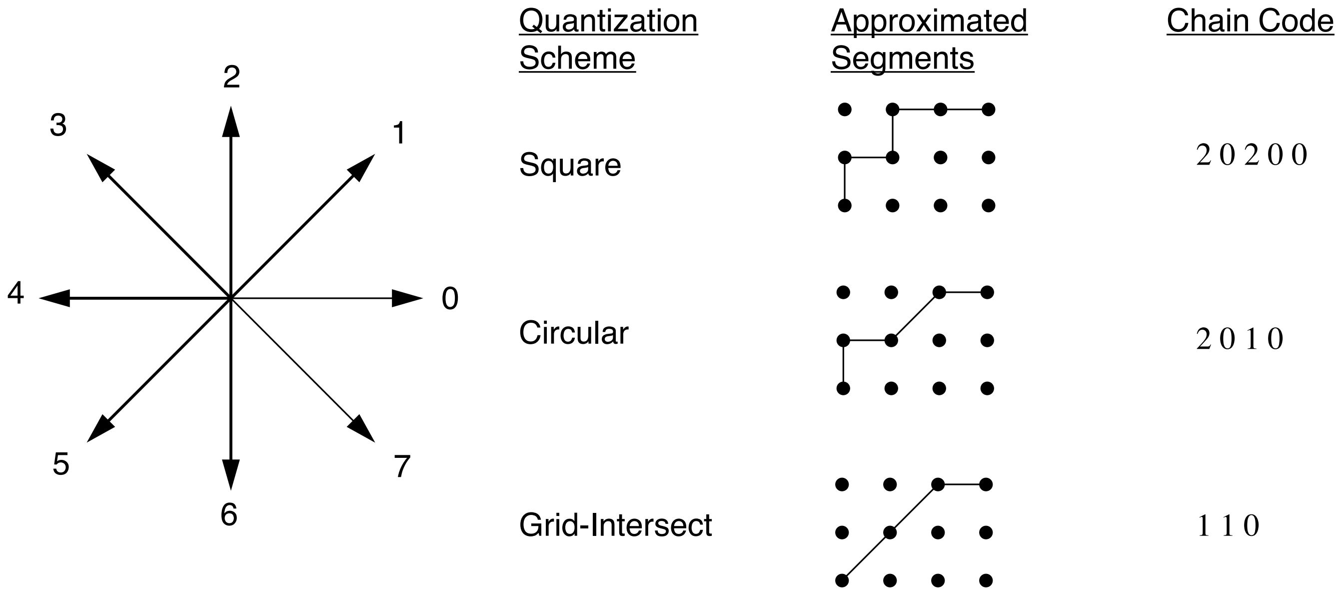


Fig. 1 Illustration of “square”, “circular” and “grid-intersect” quantization. The mesh (solid lines) has a mesh-gap T.

scheme applied to the line drawing in Fig. 1 yields *abcef* as a sequence of curve points. In the third scheme, referred to as *grid-intersect* quantization, curve points are chosen on the basis of how the line drawing intersects the grid line between two adjacent mesh nodes. Of the two mesh nodes, the one closer to the intersection point is chosen as a curve point. Thus in Fig. 1, *a* rather than *b* is chosen as the curve point because the distance *ra* is smaller than the distance *rb*. Furthermore, only a single curve point is chosen when more than one intersection point lies in the neighborhood of one mesh node. This procedure yields *acef* as a sequence of curve points for the line drawing in Fig. 1.

Once a quantization scheme is specified, the line drawing can then be approximated by segments joining the curve points. From a given curve point in the sequence, the next curve point lies at most in one of eight directions, as illustrated in Fig. 2 (a). Hence the approximated line drawing can be efficiently coded using three bits per segment. The approximating segments, along with their chain codes, that result when one applies the three different quantization schemes to the line drawing depicted in Fig. 1 are illustrated in Fig. 2 (b). Note that they yield different chain codes. In particular the chain codes differ with respect to the number of *diagonal* elements they contain. Square quantization, by its nature, does not yield any diagonal elements in the chain code. We are assuming here that the curve does not pass through the intersection point of two mesh lines in which case there is an ambiguity with respect to the curve points to choose. However, in a random curve or line drawing the probability that it intersects such a point is zero. Freeman has argued that, in order to “best” approximate the true curve, diagonal elements should occur about 50% of the time for random configurations. This design factor spurred a search for other quantization schemes such as circular, and grid-intersect quantization which, as Fig. 2 illustrates, do yield chain codes with diagonal elements. In [5] Freeman derives values for the probability of obtaining diagonal el-



(a) (b)

Fig. 2 Chain encoding a curve with three quantization techniques.

ements, $Pr(d)$, with circular and grid-intersect quantization.

In the following sections a family of quantization schemes, referred to as p-Minkowski metric quantization is introduced. Under the assumption that the mesh-gap is small relative to the curvature of the line drawing, for p=1,2 and infinity p-Minkowski metric quantization reduces to grid-intersect, circular, and square quantization, respectively, thus unifying the three approaches. This is so because for a sufficiently small mesh-gap the curve can be considered locally to be a straight line. A theoretical analysis using geometrical probability theory is illustrated for the case of circular quantization along the lines given in [13]. Some additional topics regarding the processing of chain codes are also touched upon.

2. Minkowski Metric and Convex Quantization

In the introduction we stated that curve points are those nodes which lie “close” in some sense to the given line drawing. In this section we give a more formal definition of curve point in a general framework which leads to two families of quantization schemes.

Given two points $x=(x_1, x_2)$ and $y=(y_1, y_2)$, a general measure of their closeness is the p-Minkowski metric given by

where p is a positive real number such that $p \geq 1$. For p=2, $d^2(x,y)$ is the well known Euclidean distance. These metrics have found wide application in communication theory [6], picture processing [7], information theory [8], decision theory [9], and discriminant analysis [10]. For $p = \infty$ the

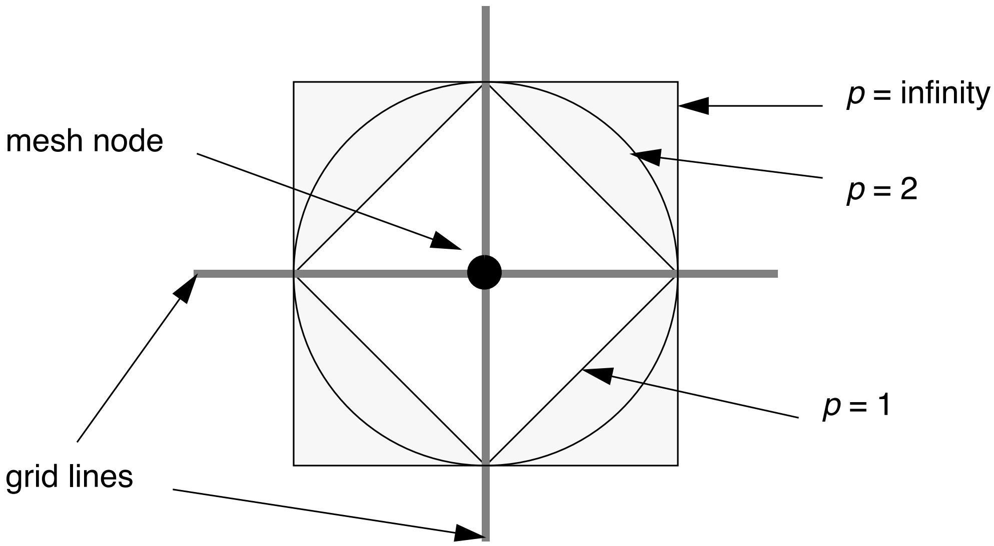


Fig. 3 Illustrating the θ-quantization regions for different values of θ.

$$d^p(x, y) = [|x_1 - y_1|^p + |x_2 - y_2|^p]^{\frac{1}{p}}$$

above equation reduces to

$$d^\infty(x, y) = max[|x_1 - y_1|, |x_2 - y_2|]$$

Fig. 3 shows the loci of constant distance $d^p(x,y) = T/2$ for three values of p: p=1,2,∞, centered about a mesh node θ serving as the origin. Note that the loci determine regions or neighborhoods about the mesh node. Let L denote the set of points that constitutes the line drawing. The distance of a mesh node θ to the set L is defined as the distance from θ to the closest point in L. In other words, for a given metric d^p we define

$$d^p(\theta, L) = inf\{d^p(\theta, x), x \in L\}$$

Definition: For a given line drawing L and mesh gap T a mesh node is a *curve point*, according to p-Minkowski metric quantization, if, and only if, $d^p(\theta, L) < \frac{T}{2}$.

For p=∞ and p=2 the above quantization scheme reduces to square and circular quantization. For p=l we have a new quantization scheme which we could call "rhombic" quantization. Sometimes in practice and particularly in the theoretical analysis, T is chosen small enough such that the line drawing is essentially a straight line within the square determined by consecutive vertical and horizontal mesh likes. When this is true a little thought will reveal that for p=l we have in

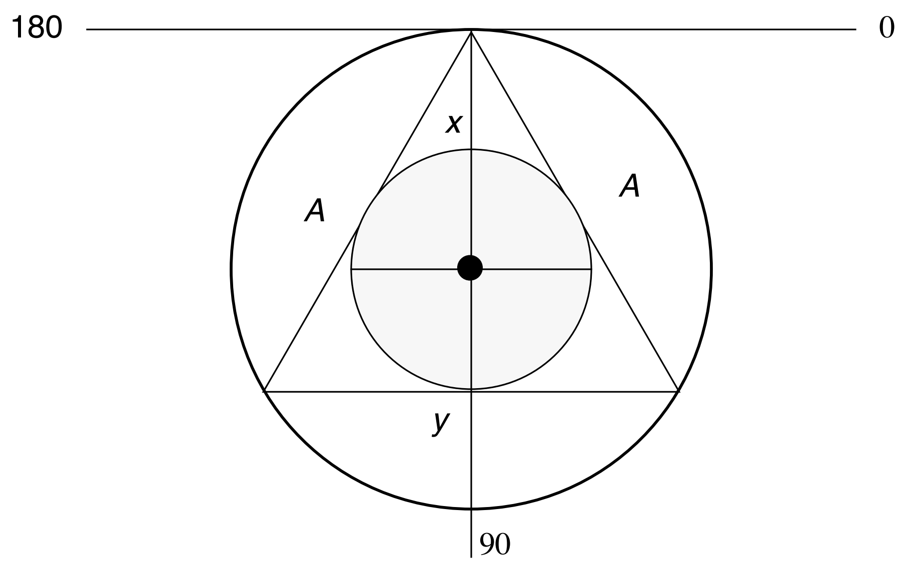


Fig. 4 Illustrating several definitions of random chords.

fact grid-intersect quantization.

In Fig. 3 the value of p determines a neighborhood about θ whose boundary lies in the shaded region. A more general family of quantization schemes can be defined by specifying a neighborhood about θ with an arbitrary boundary in the shaded region as long as it is convex. This latter scheme will be referred to as *convex* quantization. In the next section we turn our attention to determining the probability of obtaining diagonal elements in chain coding for the general quantization schemes discussed above.

3. Probability of Obtaining Diagonal Elements

In order to optimize the efficiency of Freeman encoding it is of interest to determine, for a given quantization scheme, what the probability of obtaining diagonal elements is for arbitrary or "random" configurations in the long run, after it has executed on a large number of line drawings. In order to analyze the problem theoretically we need a model of the process of obtaining a chain code for an arbitrary line drawing. The model used by Freeman [5] and here is one such that if we were to pick, out of all possible line drawings, a random square of side T, such as *cdeg* in Fig. 1, through which the line drawing passes, the line drawing within that square can be considered without loss in accuracy as a random straight line. With this model, in order to find the probability of obtaining a diagonal element $Pr(d)$ for a given quantization scheme, it suffices to find the probability that a random straight line within a square such as *cdeg* passes through the corresponding quantization regions. However, one must be careful when invoking a theoretical model for random lines as one can easily fall into the trap of the well known *Bertrand* paradox. This paradox will be illustrated with the same problem used in Duda and Hart [7]. The problem is the following. Suppose we have a circle of unit radius. What is the probability that a random chord of the circle has a length greater than the side of an inscribed equilateral triangle.

First solution

Choose one end of the chord to lie on any arbitrary point on the circumference of the circle. By symmetry and with no loss of generality we can construct a triangle such that the above end of the chord coincides with a vertex of the triangle, such as the upper vertex shown in Fig. 4. To construct a chord we can "project a beam of light" from the vertex to the circle at any angle θ such that $0 \leq \theta \leq 180$. Clearly only the chords obtained for $60 \leq \theta \leq 120$ will be greater than the side of the triangle. Hence, the resulting probability is precisely 1/3.

Second solution

Again, by symmetry and with no loss of generality we can assume the chord has a horizontal orientation. Clearly, in order for it to be longer than the side of the triangle, it must cross the vertical diameter somewhere between x and y. Hence, the resulting probability must be 1/2.

Third solution

We can construct a chord by choosing a point at random inside the circle and then drawing a chord through this point and perpendicular to the line joining this point to the center of the circle. A chord is thus associated with every point in the circle. If we choose those points falling inside the circle of radius 1/2 then the resulting chords will be longer than the side of the triangle. This probability is the ratio of the areas of the two circles and is therefore equal to 1/4.

Fourth solution

Let us return to solution number 1 where one end of the chord lies at the vertex of the triangle. We can complete the chord by choosing a point at random in the circle and drawing the chord from the vertex through this point. Clearly, if a point is chosen in either segment A or B the resulting chord will be shorter than the side of the triangle. Hence, the resulting probability must be equal to the area of the circle, less the area of $A + B$, all divided by the area of the circle. This probability equals $1/3 + \sqrt{3}(\pi/2) = 0.61$.

We have exhibited solutions ranging from 0.25 to 0.61. What is the "true" probability? The source of the paradox lies in the ambiguous term "random line." In order to define a random line precisely we must define the line in terms of some parameters and then we must specify probability distributions on those parameters. Different parameterizations yield different results. You may ask yourself which of all these is the "correct" parameterization? In order to extricate ourselves from this dilemma we appeal to the principle of invariance. For a full treatment of this principle the reader is referred to books on measure theory and geometrical probability theory. The basic idea is the following. Suppose we throw a random line on this page and that this page contains a figure on it somewhere, say a square. We ask what is the probability that the random line intersects the square. It is intuitively satisfying that we demand that for true random lines this probability should be the same even if we change the location or orientation of the square. In other words, we demand that our probability be invariant under rotations and translations of figures. It turns our that only one definition of a random line satisfies these invariance properties. This definition involves the "normal" parameterization of a line with parameters ρ and θ where θ is the angle of a normal from the origin to the line and ρ is the length of the normal. Furthermore, ρ and θ are independent random

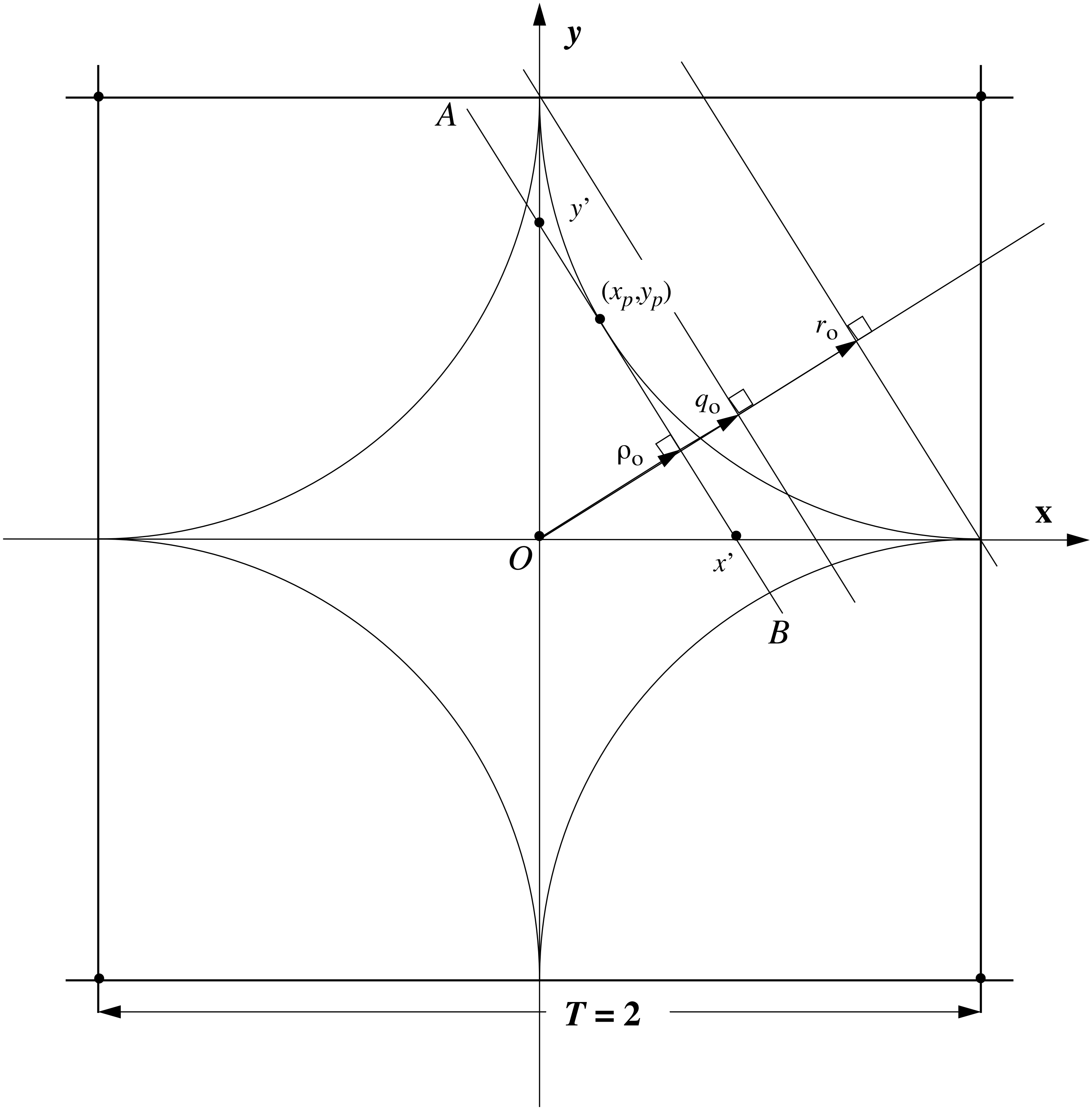


Fig. 5 Computing the probability of diagonal elements for grid-intersect quantization.

variables distributed uniformly over their relevant domains. Freeman's [5] analysis and results are as a matter of fact incorrect because in his analysis ρ was held fixed at a value of zero rather than allowed to vary uniformly over its relevant domain. We now illustrate the correct analysis for the case of circular quantization.

Consider the square region determined by four mesh nodes as in Fig. 5 where T=2 in order to simplify analysis. Since the problem is symmetrical about vertical, horizontal and the diagonal axes determined by $x+y$ and x-y, we need only consider the range $0 \le \theta \le \pi/4$. For any given value of $\theta = \theta_0$ all possible relevant lines are specified by $0 \le \rho \le r_0$. This is true because segments are either diagonal or non-diagonal and hence they must involve at least two mesh nodes. If ρ were greater than r_0 only one mesh node would be a curve point. Furthermore, all random lines yielding diagonal elements are specified by $0 \le \rho \le \rho_0$. In addition, random lines yielding two non-diagonal elements are specified by $\rho_0 \le \rho \le q_0$, and random lines yielding a single non-diagonal element by $q_0 \le \rho \le r_0$. Therefore, for a given value of $\theta = \theta_0$ the probability of obtaining a diagonal element is $Pr(d/\theta) = \rho_0/(\rho_0 + 2(q_0 - \rho_0) + (r_0 - q_0))$, the first interval yields one diagonal element, the second interval yields two non-diagonal elements (a vertical and a horizontal element) and the third interval yields a vertical element. Simplifying this expression yields $Pr(d/\theta) = \rho_0/(r_0 + q_0 - \rho_0)$. Since r, q, and ρ are functions of θ we must find $r(\theta)$, $q(\theta)$, $\rho(\theta)$ and integrate over θ to obtain a total measure for ρ_0, r_0 and q_0.

Since the radius of the circle is unity, we have

$$r(\theta) = \cos\theta. \tag{1}$$

Similarly,

$$\rho(\theta) = y' \cos(\pi/2 - \theta), \tag{2}$$

where y' is to be determined. The equation of the line AB is

$$y = -1/(\tan\theta)\, x + y'. \tag{3}$$

Since the point (x_p, y_p) is on this line, we can solve for y' if we have x_p and y_p. Now, the point (x_p, y_p) lies on the circle

$$(1-x)^2 + (1-y)^2 = 1. \tag{4}$$

The slope at (x_p, y_p) is given by

$$dy/dx = -1/(\tan\theta). \tag{5}$$

But from (4) we also have

$$dy/dx = -(x-1)/(y-1). \tag{6}$$

Combining (5) and (6) we have

$$(1\text{-}y)/(1\text{-}x) = \tan\theta. \tag{7}$$

Solving simultaneously for x and y in (4) and (7) yields

$$x_p = 1 - \sqrt{\frac{1}{1 + (\tan\theta)^2}} = 1 - \cos\theta, \tag{8}$$

$$y_p = 1 - \sqrt{\frac{(\tan\theta)^2}{1 + (\tan\theta)^2}} = 1 - \sin\theta. \tag{9}$$

Substituting (8) and (9) for x and y in (3) and solving for y' yields

$$y' = 1 - \sqrt{\frac{(\tan\theta)^2}{1 + (\tan\theta)^2}} + 1/(\tan\theta) - 1/(\tan\theta)\sqrt{\frac{1}{1 + (\tan\theta)^2}} = 1 - (1 - \cos\theta)/(\sin\theta)$$

and hence

$$\rho(\theta) = [1 - (1 - \cos\theta)/(\sin\theta)]\cos(\pi/2 - \theta). \tag{10}$$

Furthermore,

$$q(\theta) = \cos(\pi/2 - \theta). \tag{11}$$

It follows that the probability of obtaining a diagonal element is given by

$$Pr(d) = \left(\int_0^{\frac{\pi}{4}} \rho(\theta)d\theta\right) \Big/ \left(\int_0^{\frac{\pi}{4}} r(\theta)d\theta + \int_0^{\frac{\pi}{4}} q(\theta)d\theta - \int_0^{\frac{\pi}{4}} \rho(\theta)d\theta\right). \tag{12}$$

Substituting (1), (10), and (11) into (12) and integrating yields

$$Pr(d) = 4/\pi - 1 \cong 0.275.$$

A similar procedure can be followed with Minkowski quantization but the resulting expression is unwieldy and not very useful as it contains integrals which cannot be solved until the metric is specified.

A similar procedure applied to grid-intersect quantization yields a probability equal to

$$Pr(d) = \sqrt{2} - 1 \cong 0.414.$$

A careful consideration of the problem suggests that this is the largest that the probability of obtaining diagonal elements can be with any quantization scheme of this form, and suggests that it is fruitless to search for a method that will yield a probability equal to 0.5. In fact, if one considers that a random line drawing should be approximated by diagonal and non-diagonal elements equally in terms of length, then, since a diagonal element is $\sqrt{2}$ times the length of a non-diagonal element, one can argue that diagonal elements should occur with probability $T/(T + \sqrt{2}\ T)$ rather than 0.5. Now, $T/(T + \sqrt{2}\ T)) = \sqrt{2}$ -1, which is precisely the probability obtained by grid-intersect quantization. In this sense grid-intersect quantization can be said to optimally allocate diagonal and non-diagonal elements to a chain code approximation of a line drawing.

4. Chain Difference Encoding

We have seen that a chain of a line drawing is a concatenation of integers representing the directions of the elements. More formally, given a curve A, the approximating curve A' is called the *chain of A* and we write

$$A' = a_1 a_2 ... a_n = C_{i=1}^{n} a_i$$

where a_i is the ith element of the chain. Since a_i can take on eight different values as shown in Fig. 6 each element requires three bits of storage. This is how efficiency of storage is realized. A question which comes to mind at this point is: can we store a chain code using less than three bits per element? The answer is yes, but before going into the method (chain difference encoding) let us backtrack to Fig. 4.

Most of the relevant information in patterns or images is believed to be contained in the contours of those patterns. Furthermore, those parts of the contour that have high curvature are in turn considered most important. This point is illustrated by Fig. 4a. Attneave [14] - [15] asked subjects to mark the points considered important on a picture of a cat and joined a set of these points with straight lines. The resulting figure is readily identified as a sleeping cat. This suggest that the straight lines in the line drawing of Fig. 4a are not too important in the sense that they are redundant where they do not change direction. In fact if we delete parts of the line drawing as we did in Fig. 4b we can still readily identify a sleeping cat. Spitz and Borland [16] concluded from experiments of this type that line drawings of familiar objects are about 50% redundant as measured by the recognition levels maintained after varying percentages of the lines have been deleted. Chain difference encoding exploits this redundancy.

Redundancy in line drawings manifests itself in the chain through the fact that, given an element a_i, the next element in the chain does not in general assume its 8 possible values with equal probability. Most probably the next element has the same angle or it changes $\pm 45^{o}$. This means that we do not have to code the direction of each element. All we have to do is code the direction of the first element and subsequently only code the changes in direction. We can accomplish this by the following list of code symbols along with their intended meaning or message. The symbols **+** and - superscripted with an 8 denote addition and subtraction, respectively, modulo 8.

CODE SYMBOL	MESSAGE
0	$a_{i+1} = a_i$

1	$a_{i+1} = a_i +^8 1$
2	$a_{i+1} = a_i -^8 1$
3	amplifier

when $a_{i+1} = a_i$ we have no change on slope. When $a_{i+1} = a_i +^8 1$ the slope increases by 45^o, and so on. The amplifier (code symbol "3") augments the next code symbol's effective magnitude by one. In other words,

31 implies $a_{i+1} = a_i +^8 2$ $(+ 90^o)$

32 implies $a_{i+1} = a_i -^8 2$ $(- 90^o)$

331 implies $a_{i+1} = a_i +^8 3$ $(+ 135^o)$

etc.

Clearly our code symbols take on only four values now and as long as changes are $\pm 45^o$ we need only two bits per element with three bits for the first element. Once in a while, when a change greater than $|45^o|$ occurs we need the amplifier and the average number of bits required will increase by a small fraction. In conclusion, chain difference encoding requires $(2 + \varepsilon)$ bits of storage per element. As an example consider the grid-intersect chain code in Fig. 7 - 110 (9 bits). With chain difference encoding we would obtain 102 or 7 bits.

5. Ambiguity Aspects of Chain Representation

Chains have both a *spatial* and a *temporal* sense. Temporal information is useful for *dynamic* pattern recognition. Examples of dynamic of dynamic patterns are: weather, ocean currents, and migration patterns of birds. Sometimes even character recognition can be partly dynamic as in the problem of on-line character recognition.

For *static* patterns the chain code introduces ambiguities. Consider a letter "*L*" to have a chain code

$$L_1 = 66666000.$$

Another representation of L is

$$L_2 = 44422222.$$

L_1 is said to be the inverse of L_2 and we write $L_2 = L_1^{-1}$. In terms of elements, $a_i^{-1} = a_i +^8 4$. Therefore we have

$$[C^n_{i=1} a_i]^{-1} = C^1_{i=n} a_i^{-1}.$$

Every simple, open line drawing has 2 chain representations. A closed line drawing has $2n$ chain representations.

6. Some Properties of Chains

1. Chains are invariant to translation.

2. Rotation by $45k^{o}$ is accomplished by addition, mod-8, of k to each element. Rotation is distortion-free if k is even. If k is odd then elements of length T become $\sqrt{2}T$ and vice-versa.

3. The length L of a chain equals $TN_e + \sqrt{2}TN_o$ where N_e and N_o are the number of even and odd elements, respectively. For large n we have

$$L \cong nT\,(1 + 0.414\rho),$$

where ρ is the probability of obtaining diagonal elements and, as we saw earlier, is a function of the quantization scheme employed.

Let a_{xi} and a_{yi} be the normalized x and y coordinated of a_i. In other words

a_i	a_{xi}	a_{yi}
0	1	0
1	1	1
2	0	1
3	-1	1
4	-1	0
5	-1	-1
6	0	-1
7	1	-1

4. The *width W* and *height H* of a chain $a_1a_2...a_n$ are given by

$$W = (W_i)_{max} - (W_i)_{min}$$

$$H = (H_i)_{max} - (H_i)_{min}$$

where $i = 0, 1, 2,..., n$,

$$W_i = \sum_{j=1}^{n} a_{xj}, \mathrm{H_i} = \sum_{j=1}^{n} a_{yj},$$

and $W_o = H_o = 0$.

5. if $W_n = H_n = 0$ the chain is closed.

6. If there exists a pair of values of i, say (r, s) such that $W_r = W_s$ and $H_r = H_s$, the chain has a self-intersection at the point whose x and y coordinates, relative to the starting point of the chain are W_r and H_r, respectively.

7. Given a line contour which is single-valued in x, the area between the contour and the x axis is

$$\text{Area} = \sum_{i=1}^{n} a_{xi} \ (H_{i-1} + (1/2)\ a_{yi}).$$

7. Chain Correlation Functions

Sometimes shaped have to be clustered into groups and at other times a given shape has to be classified into one of several classes of shapes. In these situations we need to extract some information from the chain that characterizes the shape. Correlation functions are useful tools for extracting this information. Suppose we want to measure the similarity between a given chain $a = C^n_{i=1} a_i$ and a prototype chain $b = C^n_{i=1}\ b_i$, and suppose a and b have the same length and orientation. The *chain cross-correlation* between a and b is defined as

$$\varnothing_{ab} = 1/n \ \sum_{i=1}^{n} \cos[a_i b_i]$$

where $\cos\,[a_i\ b_i]$ is defined as the cosine of the angle between elements a_i and b_i. Note that for identical chains $\varnothing_{ab} = 1$. More generally the two chains will be of different length. In this case we can slide one chain along the other in order to obtain several values of $\varnothing_{ab}$ which define a correlation function. Suppose we have two chains of different lengths, $a = C^n_{i=1} a_i$ and $b = C^m_{i=1} b_i$, $n < m$. Accordingly, we define the chain *cross-correlation function* as

$$\varnothing_{ab}(j) = 1/k \ \sum_{i=1}^{n} \cos[a_i b_{i+j}]$$

where k is the length of the intersection of the two sequences. Alternately we can define

$$\varnothing_{ab}(j) = 1/n \ \sum_{i=1}^{n} \cos[a_i b_{i+j}]$$

where $i+j$ is modulo m. If one value is desired, as for purposes of classification, the maximum value of $\varnothing_{ab}(j)$ can be chosen. If $n = m$ we have $\varnothing_{ab}(j) = \varnothing_{ba}(-j)$. Furthermore, if a=$b$ (i.e., we only have one chain) $\varnothing_{aa}(j)$ becomes the *chain auto-correlation function* which characterizes the chain, and hence the original line drawing, and can be used for the purposes of clustering. As a final comment it should be noted that these correlation functions cannot be expected to be adequate unless the original curves are of the same scale and orientation.

8. References

[1] H. Freeman and S. P. Morse, "On searching a contour map for a given terrain elevation profile," *Journal of the Franklin Institute*, Vol. 284, No. 1, July 1967, pp. 1 - 25.

[2] H. Freeman, "Computer processing of line-drawing images," *Computing Surveys*, Vol. 6, No. 1, March 1974, pp. 58 - 97.

[3] H. Freeman and J. M. Glass, "On the quantization of line-drawing data," *IEEE Transactions on System Science and Cybernetics*, Vol. SSC-5, January 1969, pp. 70 - 79.

[4] H. Freeman, "On the encoding of arbitrary geometric configurations," *IRE Trans. Electronic Computers*, Vol. EC-10, pp. 260 - 268, June 1961.

[5] H. Freeman, "A technique for the classification and recognition of geometric patterns," *Proc. 3rd International Congress on Cybernetics*, September 1961, pp. 348 - 369.

[6] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*, John Wiley & Sons, Inc., New York, 1965, pp. 459 - 460.

[7] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley, 1973, pp. 348 - 350.

[8] G. T. Toussaint, "Distance measures as measures of certainty and their application to statistical pattern recognition," presented at *Conf. on Theoretical and Applied Statistics and Data Analysis*, Queen's University, Kingston, Ontario, June 4 - 6, 1973.

[9] G. T. Toussaint, "On a simple Minkowski metric classifier," *IEEE Trans. Systems Science and Cybernetics*, Vol. SSC-6, October 1970, pp. 360 - 362.

[10] R. S. Chikara and P. L. Odell, "Discriminant analysis using certain normed exponential densities with emphasis on remote sensing applications," *Pattern Recognition*, Vol. 5, No. 3, September 1973, pp. 259 - 273.

[11] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 1965.

[12] M. G. Kendall and P. A. P. Moran, *Geometrical Probability*, Charles Griffin & Co., London, 1963.

[13] J. Koplowitz and G. T. Toussaint, "A unified theory of coding schemes for the efficient transmission of line drawings," *Proceedings of the 1976 IEEE Conference on Communications and Power*, 20 - 22 October, Montreal.

[14] F. Attneave, "The quantitative study of shape and pattern perception," *Psychological Bulletin*, Vol. 53, 1956, pp. 452 - 471.

[15] F. Attneave, "Some informational aspects of visual perception," *Psychological Review*,

Vol. 61, 1954, pp. 183 - 193.

[16] H. H. Spitz and M. D. Borland, "Redundancy in line drawings of familiar objects: effects of age and intelligence," *Cognitive Psychology*, Vol. 2, 1971, pp. 196 - 205.