and a discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, pp. 90–93, Jan. 1974.

[4] K. S. Fu, *Sequential Methods in Pattern Recognition and Machine Learning.* New York: Academic, 1968.

[5] H. C. Andrews, "Multidimensional rotations in feature selection," *IEEE Trans. Comput.*, vol. C-20, pp. 1045–1051, Sept. 1971.

[6] H. C. Andrews, *Introduction to Mathematical Techniques in Pattern Recognition.* New York: Wiley-Interscience, 1972.

[7] S. Randys and V. Pikelis, "On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, pp. 242–251, May 1980.

Manipulation and Presentation of Multidimensional Image Data Using the Peano Scan

R. J. STEVENS, A. F. LEHAR, AND F. H. PRESTON

***Abstract*—A fractal curve first discussed by Giuseppe Peano has useful statistical properties in image processing. The curve can act as a transform between a line and *n*-dimensional space retaining some of the spatially associative properties of the space. Applications of the curve include multispectral image display, compression in the color and spatial domain, classification, and color display. The technique enables the display of high quality color images on a frame store capable of displaying only 256 colors simultaneously. A logically ordered and complete set of colors for false color work can be selected. An improvement in compression by about 3× for color images can be obtained by using this method in conjunction with an existing spatial compression technique. A simple algorithm for constructing the curves is shown.**

***Index Terms*—Data ordering, false color, image coding, image processing, multispectral sensing.**

SUMMARY

A technique has been developed for adaptively reordering and compressing multidimensional image data to facilitate classification, analysis, or presentation of information. In this paper, a Peano curve is used to reorder three band image data to and from a single dimension. A variety of techniques already available for single channel data can then be applied because the transform preserves adequately many spatial properties of the original data. The immediate use, for which the technique was developed, is for the presentation of high color-fidelity images on a frame store able to display only 256 colors simultaneously, by choosing the colors displayed to match those of the original image. The method is unusual in that reordering and compression is performed in the color rather than the spatial domain, although the process orders data ideally for the further use of many standard compression techniques. For color images, an improvement in compression factor by 3× will be possible by using this method in conjunction with existing spatial compression techniques. Further applications in false color presentation, multispectral imagery, classification, and color printing are described. Algorithms for generating the curves are outlined in the Appendix.

Manuscript received April 5, 1982; revised July 14, 1982.

The authors are with the Scientific Research and Development Branch, Home Office, Woodcock Hill Sandridge, St. Albans, Hertfordshire AL4 9HQ, England.

INTRODUCTION

Multidimensional data are commonly encountered in many signal processing tasks and present problems in display and interpretation. Many analytic techniques used to process the data are essentially single channel operations. Examples include histogram equalization, adaptive thresholding, and many compression techniques such as run length encoding or delta modulation. Similarly, the three primary colors red, green, and blue used to form color images are often processed separately until they are recombined for a color display.

If linear analytic techniques are to be applied, there is therefore a requirement for a means of converting the *n*-dimensional data into one dimension. The obvious method is some form of raster scan, which covers the data space. However, there are superior methods of generating a track through *n*-dimensional space which preserve some of the statistics of the spatial properties of the data. These involve the use of Peano scans, which act a transform from a line to *n*-dimensional space, and vice versa.

PEANO SCANS

In 1890 the Italian mathematician Giuseppe Peano [1] discovered a family of curves which scan *n*-dimensional space. These curves preserve some of the spatial associativity of the scanned dataspace on the single dimension formed by the scan. There is little spatial movement for a large number of steps along the curve, because the curve is so convoluted. Fig. 1 shows an example of a two-dimensional Peano curve, quantized to 32 × 32 resolution. The particular form of the curve used here is an orthogonal version of the set of Peano curves first demonstrated by the mathematician Hilbert, i.e., all changes of direction of the curve are whole right angles.

Fig. 2 shows a simple example of a three-dimensional Peano curve. This type of curve can be generated in a space of any dimension, and the greater the dimension of the space the more convoluted the curve. The Peano curve, and its use for pattern recognition and cluster analysis in digital space, is described by Simon and Quinqueton [2]. The curve is one of the family of fractal curves discussed in more detail by Mandelbrot [3], who singled out this curve as being suitable for practical applications. Patrick *et al.* [4] showed how similar curves could be used to map multidimensional data onto a line for different applications. Methods of generating orthogonal Peano curves are discussed in greater detail in the Appendix.

The relevant properties of the Peano scan in this application can be summarized as follows.

1) The unbroken curve passes once through every element in digital space.

2) Points close on the curve are close in space.

3) Points close in space are likely to be close on the Peano curve.

4) The curve acts as a transform to and from itself and

0162-8828/83/0900-0520$01.00 © 1983 IEEE

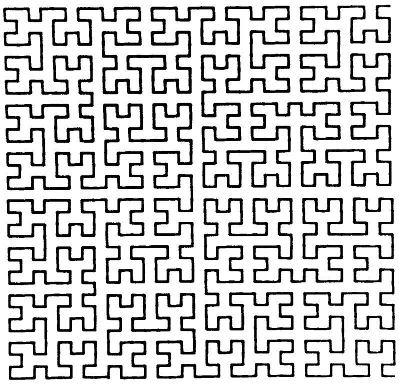


Fig. 1. Two-dimensional Peano scan.

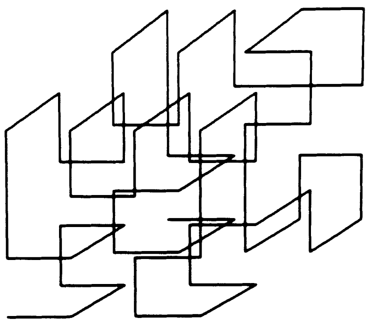


Fig. 2. Three-dimensional Peano scan.

n-dimensional space, preserving some of the spatial properties on the scan.

The space involved in this paper is quantized three-dimensional color space.

Technical Résumé

The synopsis of the technique is that the colors used in the display are picked to reflect the dominant colors of the original image. The Peano scan spans color space and orders it onto one dimension. The one-dimensional data can then be compressed logically, corresponding to a compression in color space. Firstly, a digital color space (axes red, green, and blue) is constructed and the pixel distribution in that color space plotted. The Peano scan is used to send a line through every element in that space. Because the measured distance along that Peano line is one-dimensional the pixel density distribution can then be plotted as a one-dimensional histogram. However, the Peano curves have the property of conserving some of the spatial properties of the space on the one-dimensional line. For example, should the data be clustered in space, they will be clustered along the histogram and thus the colors to be displayed can be chosen on the basis of that distribution. If pixels on the line are allocated to a close value on the line as part of a color compression, then that allocated point will represent a point close in color space. The error in color rendition will be small compared to a scanning technique which dissociates the spatial properties of the data.

Computational Aspects

Algorithms to produce two- and three-dimensional Peano curves of any required quantisation have been programmed in Fortran and were used for the examples shown in the paper. The 3-D Peano curve used spans RGB (red, green, and blue) color space where each color is resolved to five bits. The time taken to calculate 32 768 (i.e., 2^{15}) Peano coordinates on a 16-bit minicomputer is about 60 s. However, once the coordinates are known they can be stored and reproduced quickly. Typical allocation times for encoding a given image onto the calculated Peano space are about 15 min for recoding three $512 \times 512 \times 8$ bit images onto one similar image. This image may be displayed thereafter with no processing delay.

The number of colors available in the color space corresponds approximately to the spectral discrimination of the eye. When two adjacent colors on the cube with each primary color resolved to 5 bits (i.e., 2^{15} or 32 768 colors) were displayed on a color monitor, they could be just discriminated. This was very difficult at finer color resolution, i.e., the separation of two adjacent colors from 2^{18} (about 250 000) was not reliable, at least for this form of display.

Data Values and Look-Up Tables

The color space is quantized in digital display systems and the color combinations are limited by the resolution of that quantisation. If each color is quantized to 5 bits, then there are 2^{15} possible colors, while some frame stores can quantize to 8 bits per color. These two schemes represent the possibility of 32 768 and about 16 000 000 colors, respectively. Should the frame store display each of these color combinations directly, then it would require 15 and 24 bits of storage per pixel, respectively.

However, many frame stores cannot display this number of colors simultaneously because of the expense and complexity of memory and D/A converters. Moreover, the eye cannot discriminate between 16 million colors on a monitor. Often the image is stored with fewer bits per pixel but displayed with greater flexibility through a look-up table (LUT), the use of which is the key to this technique. The LUT is the programmable device which maps the data value stored in memory to a particular RGB combination on the display screen. The most flexible types of LUT are able to map out any data value to any color required, within the quantization range of both variables. For example, an 8-bit pixel data value of 137 (10001001) can be programmed to display green, red, or any other color through the LUT.

An LUT typical of those used on a well-designed frame store is able to convert any 8-bit value to any possible color quantized to 8 bits in each of the three primaries. Any one of about 16 000 000 (2^{24}) colors can be selected for a given data value but only 256 colors can be simultaneously displayed. If a complex color image is to be displayed successfully then those 256 colors must be chosen optimally for that image. For a false color image the requirement is to have maximum color discrimination and a logical color range for the data values.

The examples throughout this paper are for three-dimensional color space, quantized to 5 bits per primary color, although the Peano technique would work well for data with more dimensions, providing that processing times can be kept reasonable. Fig. 3 is a typical example of the spatial distribution of pixels in color space, for the test picture used later in the paper. The clustering of the pixels is apparent.

"Binning" the Histogram

Fig. 4 shows the two histograms for the color space of Fig. 3, generated using the raster and Peano scan methods. The clustering of data is clearly seen in the Peano approach, illustrating the retention of the spatial properties of color space. In comparison, the raster scanned data are spread over the whole of the scanned length. The problem of choosing the colors (LUT values) now becomes analogous to the well-known technique of histogram equalization. Because of the ordering along the

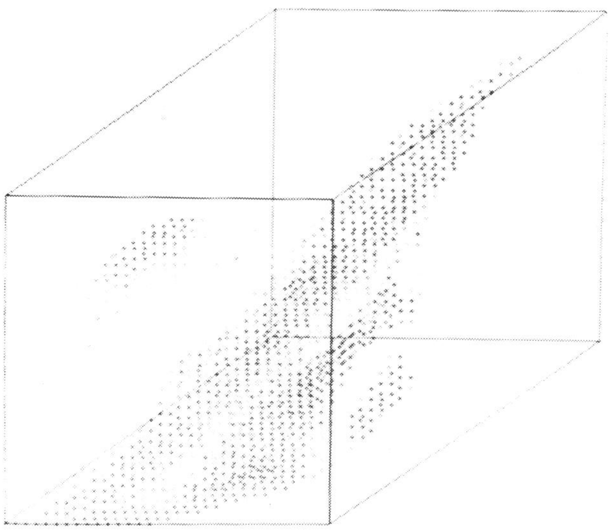


Fig. 3. Distribution of pixels in color space.

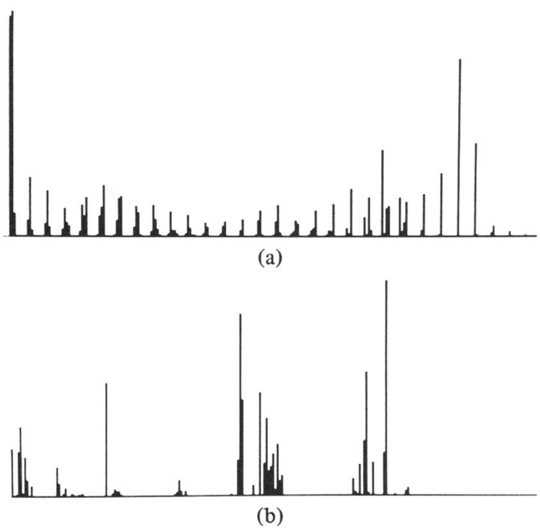


Fig. 4. (a) Raster histogram of color space. (b) Peano histogram of color space.

Peano line approximations can be made without severe color errors, as close points are close in color space. The colors are assigned by scanning along the Peano histogram and allocating the limited set of LUT values on a population density basis, i.e., "binning" the original data values.

A satisfactory method of "binning" is as follows. First, a mean "bin size" is calculated, essentially the total number of pixels divided by the total number of colors to be displayed. The histogram is scanned and all points with a population greater than a number based on this value are immediately allocated a LUT value. A new bin size is calculated for the remaining pixels and the LUT values chosen by dividing the Peano scan histogram into sections each containing approximately the same number of pixels. The half-way points between each of the LUT values were then found and points allocated to the nearest LUT value, in effect allocating all pixels to a LUT value in the center of a small volume in space defined by a length of the Peano track. This is the optimum technique so far found for images of human faces.

Fig. 5 illustrates the statistics of positional error in color space after allocation of LUT values for the image above. It plots the statistics of spatial error in color space after the color compression. The spatial error is the distance between a pixel in color space and the LUT value actually allocated to that

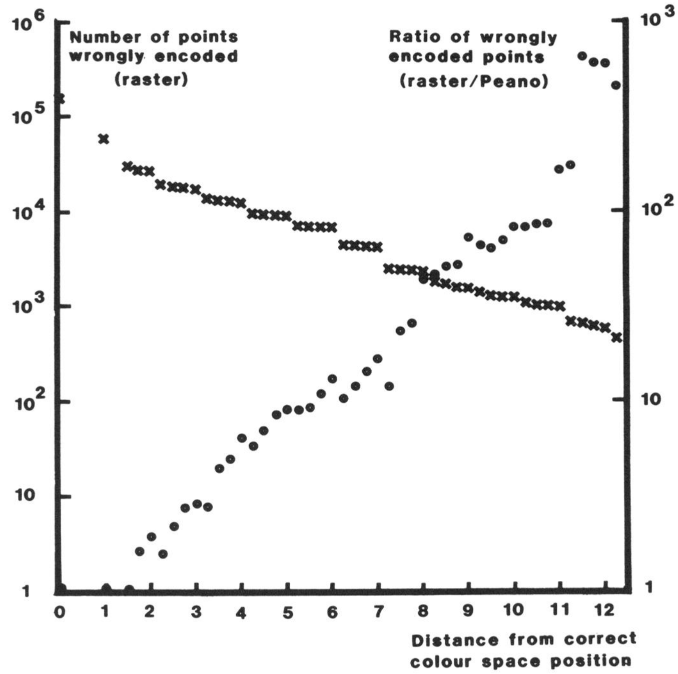


Fig. 5. Statistics of positional error in color space.

pixel. The graph shows the total number of points in error at any given distance away from their correct pixel position in color space. This number is plotted directly for the raster scan and then the ratio of this number for the Peano and raster scan is plotted versus distance. The unit of size is based on the quantization of color space into a 32 X 32 X 32 cube. For example, there are ten times as many points with a spatial error of five units for the raster scan compared to the Peano scan. The superiority of the Peano scan method is clear, especially in limiting the number of pixels that are grossly in error.

The reference picture shown in Fig. 6(a) was the picture used to derive the previous statistics. This picture was made by successive exposure of 512 X 512 X 8 bit red, green, and blue images on the monitor. An alternative technique which allows immediate display is to allocate the memory planes of the frame store directly to the primary colors. For example, 3 bits might be allocated for red, 3 bits to green, and 2 bits to blue or a similar system. The technique of allocating bit planes to primary colors is not successful as can be seen from Fig. 6(b). Contouring on the face is very pronounced because there are so few flesh tones. Most of the 256 colors available have a small or zero population and are thus wasted. The linear color strip at the base of each digital color picture is a representation of all the colors used in the picture, i.e., a display of the LUT. Data values from 0–255 are from left to right on the strip.

Picture Display with Limited Colors

Fig. 7 shows the picture of Fig. 6(a) displayed with 256 colors chosen using the binning technique described. There is no requirement to have 256 distinct values—this was chosen because of the frame store specification. The image is scarcely distinguishable from the original. On the monitor all 256 shades were distinguishable but this color discrimination will be lost in the multiple stages involved in taking the image from the monitor to the magazine page. The success on this complex image shows that the technique enables simple frame stores to be used for high color-fidelity display, with a flexible LUT. The colors in the displayed LUT correspond to the dominant colors of the original image.

The process of reducing the number of colors can be extended and eventually the method begins to break down. A face is a severe test of any technique like this because we are

(a)

(b)

Fig. 6. (a) Reference color picture. (b) Contoured color picture.

(a)

(b)

Fig. 8. (a) 50 color Peano picture. (b) 50 color raster picture.

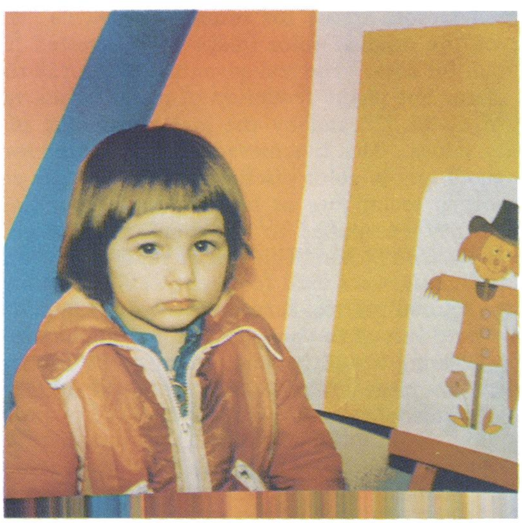


Fig. 7. 256 color picture encoded with the Peano scan.

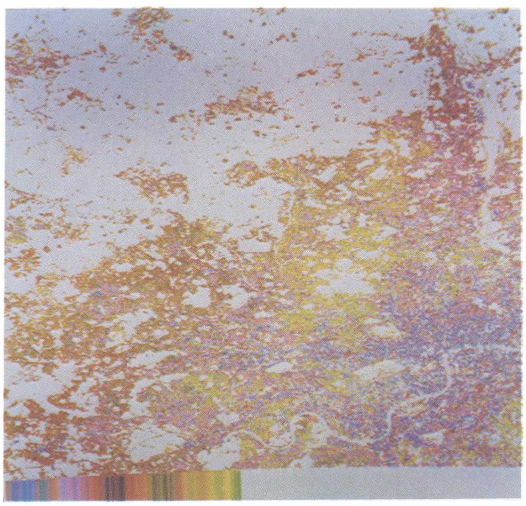
so familiar with the subtle color changes involved in skin tones and because of the distinct coloring of small areas, like eyes and lips. Fig. 8(a) shows the same picture with only 50 colors, remembering that these colors are selected by the Peano scan to be close to optimal for the colors in the original image. Contouring of flesh tones, and color error in small areas is beginning to become apparent, at least on the monitor. Nevertheless, the quality from such a limited number of colors is remarkable. Even so it is clear from the merging of some those 50 colors on the LUT that the image could be displayed with even fewer colors if the color discrimination of the human eye were taken into consideration. The method is even more successful with pictures of objects in which color is normally less predictable to the human observer. Fig. 8(b) for comparison shows the same picture scanned with a raster scan technique and the errors are far more severe. The raster technique exhibits faults which are apparent on a monitor for 256 colors. Those faults may be masked by the printing process and so the more extreme case of 50 colors is shown.

The technique enables the direct display of a high-fidelity color image with only eight or fewer bits of memory per pixel, and also enables the simultaneous storage of three times as many color images on frame stores already equipped for color imagery. Unlike many other compression techniques, no processing is involved at the time of display, although the appropriate LUT (about a thousand bytes) has to be loaded at that time.

Ordering Merits

A particular advantage of the technique is the logical ordering of the data in its encoded form. If two encoded data values for a pixel are close, then the colors represented will be

(a)



(b)

Fig. 9. (a) Satellite picture. (b) Classified satellite picture.

close in color space (but not necessarily vice versa). For example, if the pixel data is quantized to 8 bits precision (i.e., 0–255), then adjacent data values, such as 131 and 132 will represent two adjacent colors on the Peano scan. Unless the n-dimensional space has a low local population density, the two colors will therefore be close in color space. The low entropy of the data is reduced by the Peano scan in the image data when slightly different colors are allocated to the same LUT value. Hence, further compression techniques are likely to be more successful.

False Color Techniques

The Peano scan can also be used to divide a color space in order to provide a set of colors for false color presentation of data. The scan of the color space is divided into equal lengths to provide the color range. This set of colors is both logically ordered and complete, in the sense that adjacent colors are close in color space and every color in digital color space can be used. Furthermore, because the set is mapped to a line, the colors can be mapped one-to-one onto the data values. Ideally, the color space would have been arranged to reflect human color discrimination.

The colors chosen can be allocated either to a single channel of data or to data values derived from analysis of n-channel data (possibly encoded using the previously discussed method). Fig. 9(a) shows a Landsat image of London, England, recoded to $512 \times 512 \times 8$ bits from an original three band $512 \times 512 \times 6$ set of images, and given a false color Peano scan rendition as described. The false color range derived is displayed at the base of the picture. The use of this logical and complete color range brings out subtleties not apparent in other false color methods. In this case the Peano scan was truncated to remove the first part of the curve which represents dark colors, where the eye has little spectral discrimination.

The Landsat image was classified in real time into town and country areas [Fig. 9(b)]. The separation technique used was to process the LUT of the recoded image, a technique possible only because of the ordering of the data and LUT values. Similar colors can be selected and displayed while the dissimilar false colors can be blanked from the screen. If there is not sufficient color discrimination, then the whole color range could be allocated to the small part of the Peano scan of interest. This technique would be applicable for images with many more bands, as newer satellites will have.

The interesting aspect of this technique is that very little computation is involved to vary the classification parameters. This is because the processing is altering the LUT (about a thousand bytes) rather than the pixel values for each point. The real work of ordering the data has already been performed by the Peano scan.

The colors that are selected and the numbers of them selected can be varied in real time, so that the operator may visually optimize the classification. A track ball is used to control the two parameters. This process is analogous to the gray scale techniques of histogram equalization, thresholding, etc. but performed in this case on a color image.

All 256 colors of both Figs. 7 and 9(a) are discernible on the monitor. Examination of the LUT on this page enables an easy analysis of which colors cannot be distinguished from their neighbors on the color strip.

Compression

Techniques for data compression of digital images are many and varied and typical compression factors are about 5–10×. The Peano scan method is rather different to these as any compression occurs in the color domain. There is no loss of spatial resolution compared with, for example, color dither techniques. Although the data are preprocessed, there is no processing delay in reconstituting the image. Also, the data are mapped onto one-dimension with low entropy; practical compression factors are of the order of 3–4×, i.e., the three channels of the original color image are converted into one channel of 8 bit data. The real advantage lies not in this compression factor but in the reduction of possible data values by a factor of up to 2^{18} and by the logical ordering of the recoded data values. This is achieved with no spatial resolution loss and small loss in color discrimination. Further compression by normal techniques or by run length encoding a 2-D Peano scan may be applied to that data, thus increasing the overall compression factor. This will be reported separately.

If the technique were used for digital TV transmission, the processing required for real-time data encoding would be performed by hardware at the transmission end of the system and minimal processing would be required in the TV receiver. The hardware requirement at the receiver would be a LUT to control the three guns of the color set.

Color Printing

In color printing the problem is rather different. There are fewer colors available, at least for low cost work and these possible colors are fixed. It is as if the LUT is fixed. The problem then becomes the allocation of the colors in the real image to those available. The requirements may be different in different circumstances. It would be easy to use all available colors equally by binning, should this be required, or to

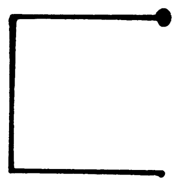


Fig. 10. Two-dimensional seed curve.

stretch out any color differences to maximize the impact of the printing, with minimum impact on the real color of the original. For example, if the color range of this journal were known, then a false color image could be adjusted to take maximum advantage of the colors available by distributing the data over that range.

Conclusions

The use of the Peano scan for mapping data logically from one dimensionality onto another would seem to have many possible applications. The result is a general purpose data ordering process, which permits classification and compression, in this case in the color domain. The technique is interesting because the initial processing orders the data, enabling the later classification, compression and display to be performed very rapidly. It is hoped that these examples of practical uses in image processing will stimulate other applications of the interesting properties of these curves.

Appendix
Generation of Orthogonal Peano Curves

The curves described in this paper are orthogonal Peano curves, one set of a family of possible types [3]. A general mathematical approach to the algorithms for generating these curves is given by Butz [5]. By contrast, a simple method of generating orthogonal Peano curves of any dimension or resolution is now shown. The general method uses a "seed" curve, which describes the overall flow of the curve through the space. This seed curve is replaced by a set of similar seed curves, some of which are rotated with respect to the original. Each of the seed curves can in turn be replaced by the same set of curves, rotated appropriately, in order to obtain the resolution required. The algorithm can be converted into simple programs which can convert a Peano length into a spatial position, and vice versa, i.e.,

$$P(1) \rightarrow x, y, z$$

or

$$x, y, z \rightarrow P(1)$$

where $P(1)$ is the length along the Peano curve. Hardware to perform the task at video rates could be constructed using the general principles outlined.

The notation used here is that S_n is the n-dimensional seed curve; for example, S_2 is a seed curve for two-dimensional Peano curves. Note that the start point of a seed is marked with a dot and that more than one seed curve is possible for a given space.

An example is given for the generation of two-dimensional Peano curves from the 2-D seed S_2. Fig. 10 is a seed curve for two-dimensional Peano curves and is replaced by the four curves shown in Fig. 11. Each of these is in turn replaced by the same set of curves in Fig. 12. This builds the basic structure of the Peano curve and the ends of the curves are linked using the seed curve which produced them. For example, the four curves of Fig. 11 are linked using the original seed curve Fig. 10 to produce Fig. 13 and those of Fig. 12 are linked by Fig. 13 to produce the completed curve shown in Fig. 14. This

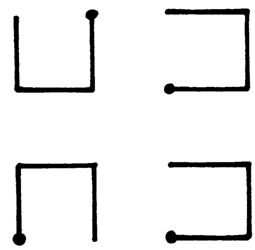


Fig. 11. Four replacement two-dimensional seed curve.

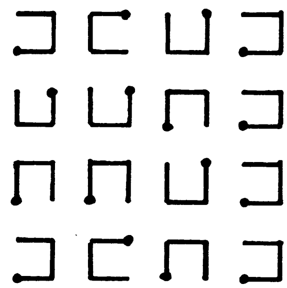


Fig. 12. Sixteen two-dimensional seed curves.

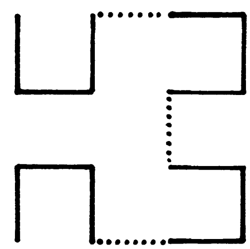


Fig. 13. Completed 2 × 2 Peano curve.

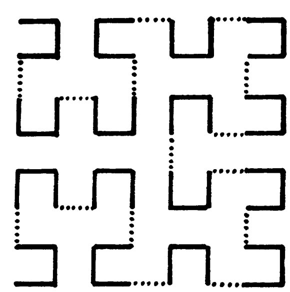


Fig. 14. Completed 2-D curve.

process can clearly be repeated indefinitely until the required quantization is reached. The software implementation is able to produce two- and three-dimensional Peano scans of any required quantization.

Generating a Seed Curve for N-Dimensional Space

An $(n + 1)$-dimensional seed (one of many possibilities) can be generated from the n-dimensional seed. It consists of the sequence

$$S_{n+1} = S_n \quad (0_1, 0_2, \cdots, 0_n, 1_{n+1}) \; - S_n \tag{1}$$

where S_n is the seed of dimension n and $-S_n$ is the inverse of that curve, each projected into space of dimension $n + 1$.

Thus the seed curve S_{n+1} can be produced from S_n by following the sequence.

1) The seed of dimension n projected in dimension $n + 1$.
2) A single step into dimension $n + 1$.
3) Retracing the n-dimensional seed.

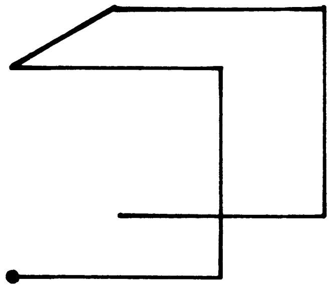


Fig. 15. 3-D seed construction.

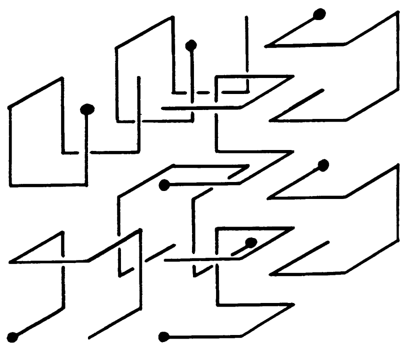


Fig. 16. Eight 3-D seeds.

The net effect of traversing the whole distance of a seed curve of dimension n is a unit step in dimension n.

A simple example of this is the generation of a three-dimensional seed curve S_3 from the two-dimensional seed curve S_2. Fig. 10 shows S_2, and by tracing S_2, a single step into the third dimension (0, 0, 1) followed by the inverse of S_2, the curve S_3 is obtained (Fig. 15).

The whole curve consists of

$$S_2 \quad (0, 0, 1) \quad - S_2.$$

i.e., (1) with $n = 2$.

The three-dimensional Peano curve is constructed by replacing the seed curve with eight similar seed curves (Fig. 16). The technique can be continued to arbitrary quantization or dimension to produce the required curve, which may contain elements of mixed dimension or resolution. The algorithm enables simple programs or hardware to generate the curves. For the computer program, there is a table of the eight possible ways of drawing the curve in two dimensions. Each of these seed curves can be described in terms of the four curves from the table that replace it when a greater quantization is required. For three dimensions, a table of 24 curves is required.

REFERENCES

[1] G. Peano, *Selected Works of Guiseppe Peano*, H. C. Kennedy, Trans. and Ed., Allen and Unwin, ISBN-0-04-164002-0.

[2] J. C. Simon and J. Quinqueton, *C. R. Acad. Sci. Paris*, vol. 286, Apr. 10, 1978.

[3] B. Mandelbrot, *Fractals–Form, Chance and Dimension*. W. H. Freeman, ISBN 0-7167-0473-0.

[4] E. A. Patrick, D. R. Anderson, and F. K. Bechtel, "Mapping multidimensional space to one dimension for computer output display," *IEEE Trans. Comput.*, pp. 949–953, Oct. 1968.

[5] A. R. Butz, "Alternative algorithm for Hilbert's space-filling curve," *IEEE Trans. Comput.*, vol. C-20, pp. 424-426, Apr. 1971.

Chain Coding with a Hexagonal Lattice

DANIEL K. SCHOLTEN AND STEPHEN G. WILSON

***Abstract*—This paper investigates the performance of chain code quantization of general curves using a hexagonal lattice structure, as a means of improving efficiency over the standard square lattice.**

Performance is first computed theoretically, assuming a generalization of grid-intersect quantization, and the curve to be quantized is assumed to be a straight line.

An algorithm is then developed to perform chain coding using the hex lattice. Computer simulations were performed to evaluate hexagonal chain coding for a variety of curves, including circles of various curvature, straight lines, and a stochastic curve model. We find that the straight-line theory is substantiated for curves whose radius of curvature is roughly twice the lattice constant. For a given peak error in quantization, hexagonal coding reduces the bit rate about 15 percent relative to the square lattice codes, and exhibits qualitative improvements in fidelity as well.

***Index Terms*—Chain codes, geometric probability, hexagonal lattice, information theory, lattice quantization region, line quantization, rate-distortion theory.**

I. INTRODUCTION

Quantization of line drawings must be performed whenever such images are to be stored or transmitted digitally. Efficient quantization minimizes the number of bits of information that is handled. When storing data, this minimizes the space required; when transmitting data, transmission time is minimized.

Chain codes are one simple way of encoding the information in a quantized curve, by recording only transitions between successive points in a lattice, not absolute coordinates. Freeman [1], [2] provides a summary of such methods.

Any regular lattice can provide chain points, but those formed by vertices of regular polygons are the simplest and most natural. While triangular and hexagonal lattices have been mentioned, only the square lattice has apparently received much attention.

Freeman [2] has studied the performance of chain codes using the standard square lattice in detail. He has analyzed three different quantization schemes for the square lattice, referred to as square quantization, circular quantization, and grid-intersect quantization. If the actual curve intersects a lattice point's quantization region, that point is added to the chain code (the transition to that point is added to the list of transitions).

Koplowitz [3] invoked results of geometric probability theory to establish a rate-distortion equation for encoding of straight lines randomly cast on the plane. Presumably, the modeling applies as well in practical cases where line drawings with small curvature are to be chain-coded.

Letting ϵ denote the maximum encoding error between the actual curve and the quantized curve, Koplowitz [3] established that for grid-intersect quantization

$$b\epsilon = 1.35 \qquad (1)$$

where b is the bit requirement in bits per unit curve length. As

Manuscript received December 16, 1981; revised May 10, 1982.

D. K. Scholten was with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22901. He is now with The Analytic Sciences Corporation, Reading, MA 01867.

S. G. Wilson is with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22901.

0162-8828/83/0900-0526$01.00 © 1983 IEEE