A New Dynamic Programming Approach for Scan Conversion of a Circle

Sushil Chandra Dimri
Professor
Graphic Era University
Dehradun- India

Sushil Kumar Chamoli
Assistant Professor
Graphic Era University
Dehradun- India

Anuj Dimri
Research Scholar
Graphic Era University
Dehradun- India

ABSTRACT

Generally curve may be generated as a sequence of many small lines; some curves like circle, parabola, ellipse, in particular can be generated with help of D.D.A. algorithm and other special algorithms. There are two main recursive algorithms for scan conversion of the circle on computer screen, the Bresenhams and the Midpoint circle generating algorithm both are pixel based; in this paper we are presenting a recursive new approach for scan conversion of the circle. The pixel in one octant has been determined with help of this algorithm and rest of the parts of the circle will be generated with help of symmetry.

General Terms

Dynamic Programming, Symmetry, Euclidean geometry, Octant

Keywords

Scan conversion, decision parameter, real theoretical point, mid point algorithm

1. INTRODUCTION

Scan conversion of continuous figure is a fundamental process in any graphics display. It is required since the raster display consists of a rectangular matrix of pixels. Scan conversion is an approximation of a given continuous figure by integer valued points (pixels) [2, 3, 9 and 10].

Circle is a very important geometrical figure. A circle is a simple shape of Euclidean geometry consisting of those points in planes that are equidistant from a given point, the centre and the distance between any of the points and the centre is called the radius.

Circles are simple closed curves which divide the plane into two regions: an interior and an exterior. In everyday use, the term "circle" may be used interchangeably to refer to either the boundary of the figure, or to the whole figure including its interior; in strict technical usage, the circle is the former and the latter is called a disk [9, 10, and 15].Scan conversion of circle received considerable research attention since 1960 [2, 12, 13, 18, 21, 22 and 23].Scan conversion of a circle means to identify the pixels (grid points) which are closest to the corresponding real theoretical points on circle [12 and 13].There are mainly to such criteria used to identify the pixels

i) Minimizing the value of function $f\ \ x, y\ \ = x^2 + y^2 - r^2$

ii) Minimizing the Euclidean distance to the circle

Circle is a symmetrical geometrical figure-1. The circle $x^2 + y^2 = r^2$ is symmetrical about x axis, y axis, line $y = x,$ and line $y = -x,$ to generate a circle on computer screen it is sufficient to find pixels on one octant and rest parts can be generated with help of reflection about x axis, y axis, line $y = x,$ and line $y = -x,$ [3, 9, 10, 13 and 14].

Cartesian coordinates: Equation of circle having centre $C\ x_c, y_c$ and radius r is given by $x - x_c\ ^2 + \ y - y_c\ ^2 = r^2.$

The parametric equation of the circle is

$$x\ = \ x_c + \ r\ cos\ \theta$$

$$y = \ y_c + \ r\ sin\ \theta$$

The equation of a circle whose center is origin and radius r is given by $x^2 + y^2 = r^2$

Its parametric equation is given by

$$x = r\ cos\theta\ \ ,\ y = r\ sin\theta,$$

Where θ is a parameter which varies from 0^0 to 360^0 [1 and 9].

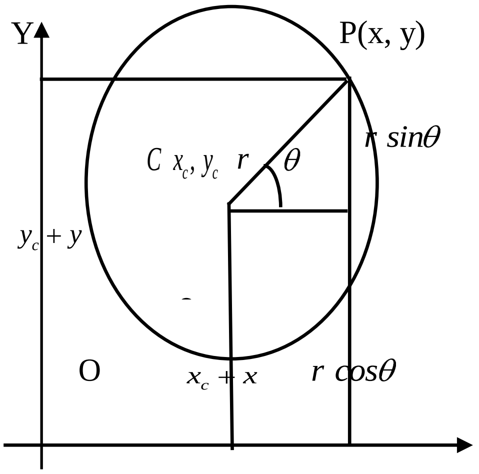

Fig 1: The circle with centre x_c, y_c

There are mainly two popular algorithms for scan conversion of circle these are Bresenham's and Midpoint circle algorithms, both the algorithms use dynamic programming approach to draw the circle. Both determine the pixel position in the octant 45^0 to 90^0, and remaining part will be generated by reflection. These algorithms use the function $f\ \ x, y\ \ = x^2 + y^2 - r^2$ to locate the pixels [2, 3 and 12].

Hsu, S. Y., Chow, L.R. and Liu, C.H. proposed an algorithm for the scan conversion of circles. They developed a new set of algorithms which they call shortest line segment incremental algorithms [16].

The similar concept has been applied to the scan conversion of line segments which is referred to as the run-length slice approach [12 and 14].

Other methods are also available which uses simple mathematical calculation and computes values of x and y separately and then rounding these values to nearest integer to plot the pixel [4 and 9].

2. PROBLEM FORMULATION

The problem is to generate a circle on computer screen which is a true simulation of a given circle $x^2 + y^2 = r^2$ using dynamic programming approach (Recursive Equation)

.

3. NEW RECURSIVE SOLUTION TO THE PROBLEM

A new dynamic programming approach for scan conversion of a circle,

Equation of a circle

$$x^2 + y^2 = r^2$$

Centre $C \equiv \ 0,\ 0$

Radius r. The parametric form of circle is

$$x = rcos\theta$$

$$y = r\ sin\theta,$$

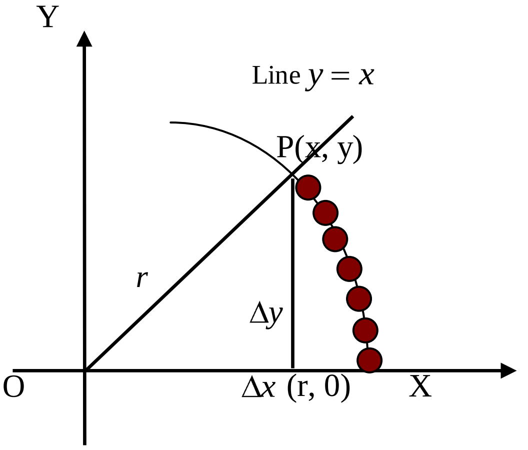

Fig 2: Pixels in octant $0 \le \theta \le 45°$

For range $0 \le \theta \le 45°$, $\Delta y > \Delta x$ change in y is larger than change in x, so at every step of pixel position determination, y will get maximum increment.

Our algorithm works differently to locate the pixel position close to real theoretical point from the Bresenhams and mid point circle algorithms; we will develop a recursive equation for decision parameter whose sign will indicate that whether x will be decremented or not.

To construct the circle we will determine the pixel position in octant $0 \le \theta \le 45°$, rest all the parts of circle will be generated with the help of reflection images about lines $y = 0,\ x = 0,\ y = x\ and\ y = -x$.

The algorithm starts from point $(r, 0)$ one by one pixel positions will be determined, here y coordinate will be incremented by 1 and decision about x coordinate will be taken on the basis of the position of next pixel position with respect to real theoretical point. To simulate the circle more accurately we will select that pixel which will be much close to real theoretical point R (shown in figure-3).

Let pixel P $x_p,\ y_p$ is to be displayed, then the next possible pixel positions are $x_p - 1,\ y_p + 1$ or $x_p,\ y_p + 1$ that pixel position will be the next one which is much close to real theoretical point R.

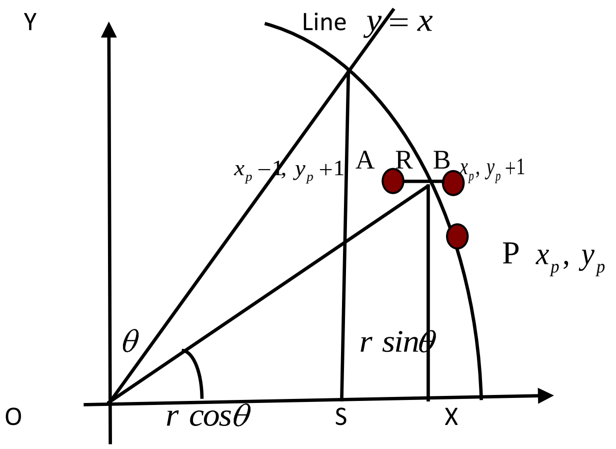

Fig 3: Position of real theoretical point R and possible pixel positions A and B

Let angle SOR is θ

Since $\Delta y > \Delta x$

$$y_{p+1} = y_p + 1$$

Now to decide whether x value will be decremented by 1 at this step or not, we will compute distances AR and RB

$$AR = r\ cos\theta - \ x_p - 1\ = d_1$$

$$RB = \ x_p - rcos\theta\ = d_2$$

Let $d_p = \ AR - RB\ = d_1 - d_2$

$$d_p = r\ cos\theta - \ x_p - 1\ - \ x_p - rcos\theta$$

$$d_p = 2rcos\theta - 2x_p + 1 \qquad (1)$$

From figure-3

$$rcos\theta = \surd\ \ r^2 - \ y_p + 1\ ^2 \qquad (2)$$

From (1) and (2) we have

$$d_p = 2\surd\ r^2 - \ y_p + 1\ ^2\ - 2x_p + 1 \qquad (3)$$

Equation (3) is a recurrence relation valid for possible values of p so

Replacing p by p+1 in equation (3) we will get

$$d_{p+1} = 2\sqrt{\ r^2 - \ y_p + 2 \ ^2} \ - 2x_{p+1} + 1 \qquad (4)$$

Subtracting equation (3) from equation (4) we have

$$d_{p+1} - d_p = 2 \ \sqrt{r^2 - \ y_p + 2 \ ^2} - \sqrt{r^2 - \ y_p + 1 \ ^2} \ - 2 \ x_{p+1} - x_p \qquad (5)$$

Or

$$d_{p+1} = d_p + 2 \ \sqrt{r^2 - \ y_p + 2 \ ^2} - \sqrt{r^2 - \ y_p + 1 \ ^2} \ - 2 \ x_{p+1} - x_p \qquad (6)$$

It is a recursive equation, of degree 1; here d_p is the decision parameter.

If $d_p < 0,$ then point A $x_p - 1, \ y_p + 1$ is close to R otherwise B $x_p - 1, \ y_p + 1$ will be close to R and so $x_{p+1} = x_p - 1$ if $d_p < 0,$

For $d_p \geq 0 \quad x_{p+1} = x_p$

Then we have if $d_p < 0$ we have

$$d_{p+1} = d_p + 2 \ \sqrt{r^2 - \ y_p + 2 \ ^2} - \sqrt{r^2 - \ y_p + 1 \ ^2} \ + 2 \qquad (7)$$

If $dp \geq 0$

$$d_{p+1} = d_p + 2 \ \sqrt{r^2 - \ y_p + 2 \ ^2} - \sqrt{r^2 - \ y_p + 1 \ ^2} \qquad (8)$$

To get the initial value of decision parameter putting $x = r,$ $y = 0$ in equation (3)

$$d_o = 2 \ \sqrt{\ r^2 - 1} \ \ - 2r + 1 \qquad (9)$$

With help of these equations, we can identify the pixel position in octant $0, \ \pi / 4$

And rest part of circle will be generated by talking reflection images of these pixels

About lines $y = x, \ y = -x, \ x = 0$ and $y = 0$.

4. PLOTTING OF GRID POINTS (PIXELS) WITH THIS NEW APPROACH:

For the circle $x^2 + y^2 = r^2$ when r =10 the algorithm plot the grid points in the octant $0 \leq \theta \leq 45°$ (figure-4), for the determined values of decision parameter using equations (7), (8) and (9).The initial value of decision parameter d_0 =0.899. It is observed that ability of this algorithm to locate the pixels is as similar to the Bresenhams and Midpoint circle algorithm.

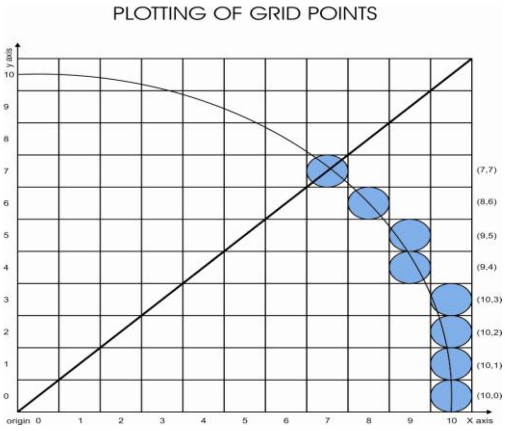

Fig 4: Plotting of pixels for $r \ = 10$ unit

Figure -4 shows the pixels determined by the algorithm also the real theoretical arc which passes through all the cells, with a little confusion for cell $(10, \ 3)$, if we apply the Mid point circle algorithm to determine the pixel at this step we will get the same result given by this new algorithm.

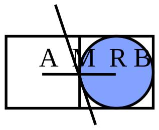

Fig 5: Position of real theoretical point R between cells $(10, \ 3)$ and $(9, 3)$.

From figure-5, $AB \ = 1, \ M$ is the mid point so $AM \ = MB = 1/2$

From figure, point R is in cell $(10, 3)$ let $\text{MR} = \text{x}$, then $BR = 1/2 \ - x$ and $AR \ = 1/2 \ + x$. Clearly $BR < \ AR$ thus point R is much close to B, Which suggest selection of pixel $B \ \ (10, \ 3)$, this matches with the result of the algorithm. For rest cells results are quite obviously matches with the results of Mid point algorithms, thus the algorithm work with the same accuracy which the existing algorithms have.

5. GENERATION OF CIRCLE WITH THIS ALGORITHM

The centre $C = \ 200, \ 200$, and radius $r = 100$

Circle generated by this algorithm on computer screen

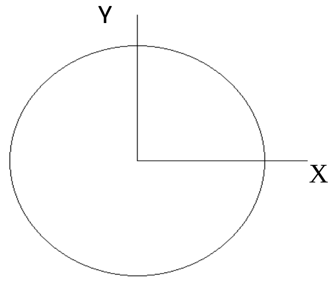

Fig 5: The circle generated by the algorithm.

6. CONCLUSION

For scan conversion of circle we has mainly two algorithms the Bresenhams algorithm and Mid point circle algorithm both these algorithm uses the function $f \ \ x, \ y \ = x^2 + y^2 - \ r^2$ to locate the next pixel position, we develop an algorithm using dynamic programming approach without using function to locate the next pixel position, which is much close to real theoretical point, this makes this algorithm different from others. In this algorithm square root involves in the calculation but the values inside the square root are very simple to compute. The time and space complexity of this algorithm and Midpoint circle algorithm is almost same. The algorithm also can further modify for scan conversion of other geometrical shapes.

7. REFERENCES

[1] Prasad, G., and Gupta, H.C., 1965 “Text Book on Coordinate Geometry” Edition-6, Pothishala, Allahabad.

[2] Bresenham, J. E., 1977 “A linear algorithm for incremental digital display of circular arcs.” Comm. ACM, vol. 20.no. 2.

[3] Foley, J. D., van Dam, A., Feiner, S. K., and Hughes, J. F., 1996 “Computer Graphics: Principles and Practice”, Second Edition in C, Addison Wesley.

[4] Gupta, S. R. and Sproull, R., 1981 “Filtering Edges for Gray-Scale Display,” SIGGRAPH 81, pp.1-5.

[5] Pang, A., 1990“Line-drawing Algorithms for Parallel Machines.” IEEE CG&A, 10, 9 (1990), pp.54-59.

[6] Sproull, R. F., 1982 “Using Program Transformations to Derive Line-Drawing Algorithms.” ACM trans. Gr., 1, 10 (1982), pp.259-273.

[7] Suenaga, Y., Kamae, T., and Kobayashi, T.,1979“A High-Speed Algorithm for the Generation of Straight Lines and Circular Arcs.” IEEE Trans. Computer, C-28, 10 (1979), 728-736.Graphics, Vol. 6 No. 1, 1 (1982), pp. 15-17.

[8] Wu, X. and Rokne, J. G. “Double-Step Incremental Generation of Lines and Circles.” Comput.Vision, Gr. Image Process. 37, 3 (1987), pp. 331-344.

[9] Hearn, D., Baker P.M., 2001 “Computer Graphics” PHI Publication New Delhi.

[10] Rogers, D., F., 1998“Procedural element of Computer Graphics” McGraw-Hill.

[11] J.F. Blinn, 1987 “How many ways can you draw a circle?” IEEE Computer Graphics and Applications, pp. 39-44, Aug., (1987).

[12] Bresenham, J.E. 1985 “Run length slice algorithms for incremental lines,” Fundamental Algorithms for Computer Graphics, R.A. Earnshaw, ed.,NATO Computer and Systems Series, vol. 17, pp. 59-104. New York: Springer-Verlag.

[13] Foley, J.D. and Dam, A.V. 1982 “Fundamentals of Interactive Computer Graphics”, Reading, Mass.: Addison-Wesley.

[14] Fung, K.Y., Nicholl, T.M. and Dewdney, A.K., 1992 “A run- length & line drawing algorithm without division operations,” Computer Graphics Forum, vol. 3, pp. 267-277.

[15] Horn, B.K.P., 1976 “Circle generator for display devices,” Computer Graphics and Image Processing, vol. 5, pp. 280-288.

[16] Hsu, S.Y., Chow, L.R. and Liu, C.H.1993 “A new approach for the generation of circles,” Computer Graphics Forum 12, vol. 2, pp. 105-109.

[17] Kulpa, Z. ‘A note on the paper by B.K.P. Horn., 1979 “Circle generators for display devices,”’ Computer Graphics and Image Processing, vol. 9, pp. 102-103.

[18] Mcllroy, M.D., 1983 “Best approximate circles on integer grids,” ACM Trans. Graphics, vol. 2, no. 4, pp. 237-263, Oct. (1983)

[19] Rokne, J.G., 1990 , Wyvill, B. and Wu, X. “Fast line scat-conversion,” ACM Trans. Graphics, vol. 9, no. 4, pp. 376-388, Oct. (1990).

[20] Van Aken J.R and Novak, M., 1985 “Curve-drawing algorithms for raster display,” ACM Trans. Graphics, vol. 4, no. 2, pp. 147-169, Apr. (1985).

[21] Wright, W.E., 1990 “Parallelization of Bresenham’s line and circle algorithms,”IEEE Computer Graphics and Applications, pp. 60-67, Sept. (1990).

[22] Wu, X. and Rokne, J., 1987 “Double-step generation of lines and circles,”CVGIP, vol. 37, pp. 331-344.

[23] Yao, C. and Rokne, G.J., 1995 “Hybrid scan conversion of Circles” IEEE Trancactions on Visualizations and Computer Graphics Vol. 1, No.4.