13. Scan Conversion Algorithms for Circle and ellipse

14 February 2024 15:31

Source: Computer Graphics by Donald Hearn and M. Pauline Baker

Properties of Circles :-

A circle is defined as the set of points that are all at a given distance r from a center position (x_c, y_c).

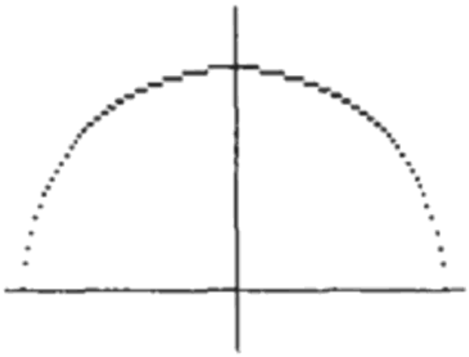
Plotting using direct methods :-

① Pythagorean theorem in Cartesian coordinates as

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

$$y = y_c \pm \sqrt{r^2 - (x_c - x)^2} \quad \text{—— (1)}$$

plotting using this equation with $(x_c, y_c) = (0, 0)$



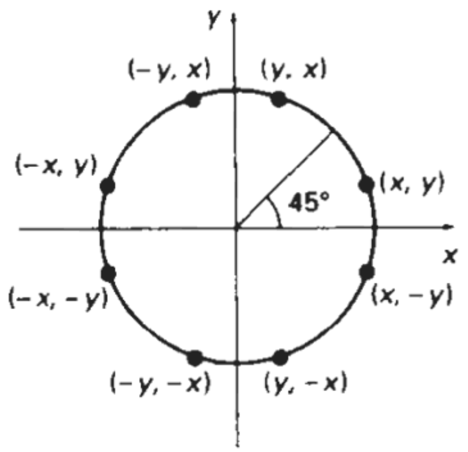
→ Spacing between the plotted pixels is not uniform.

→ Considerable computation at each step.

→ Sampling x by fixing y?

② Calculating boundary points using polar coordinates r and θ.

$$x = x_c + r\cos\theta$$
$$y = y_c + r\sin\theta \qquad — ②$$



Symmetry of a circle

Equation ① → multiplications & square root calculations

Equation ② → square root & trignometric calculations.

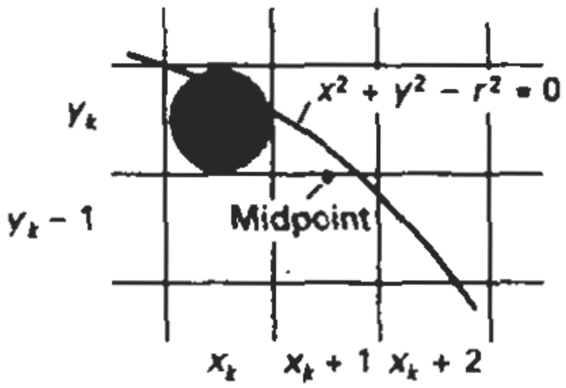
Midpoint Circle Algorithm/

Bresenham's Circle Algorithm :-

Sampling at unit intervals and determine the closest pixel position to the specified circle path at each step.

$$f_{circle}(x,y) = x^2+y^2-r^2$$

$$f_{circle}(x,y) = \begin{cases} <0, & \text{if } (x,y) \text{ is inside circle boundary.} \\ =0, & \text{if } (x,y) \text{ is on the circle boundary.} \\ >0, & \text{if } (x,y) \text{ is outside circle boundary.} \end{cases}$$



Which pixel ?

(x_k+1, y_k) or (x_k+1, y_k-1)

Decision Parameter?

Circle function $f_{circle}(x,y)$ evaluated at the mid point between the two pixels.

$$P_k = f_{circle}(x_k+1, y_k - 1/2)$$

$$p_k = f_{circle}(x_k+1,\ y_k - 1/2)$$

$$= (x_k+1)^2 + (y_k - 1/2)^2 - r^2$$

Initial decision parameter,

$$p_0 = 5/4 - r$$

$$p_0 = 1 - r \quad (\text{for } r \text{ as integer})$$

Algorithm :—

1. Input radius r and circle center (x_c, y_c), and obtain the first point on the circumference of a circle centered on the origin as

$$(x_0, y_0) = (0, r)$$

2. Calculate the initial value of the decision parameter as

$$p_0 = \frac{5}{4} - r$$

3. At each x_k position, starting at $k=0$, perform the following test :-

→ If $p_k < 0$, the next point along

the circle centered on $(0,0)$ is (x_{k+1}, y_k), and

$$p_{k+1} = p_k + 2x_{k+1} + 1$$

→ Otherwise, the next point along the circle centered on $(0,0)$ is (x_{k+1}, y_k) and

$$p_{k+1} = p_k + 2x_{k+1} + 1 - 2y_{k+1}$$

where $2x_{k+1} = 2x_k + 2$ and $2y_{k+1} = 2y_k - 2$

4. Determine symmetry points in the other seven octants.

5. Move each calculated pixel position (x, y) onto the circular path centered on (x_c, y_c) and plot the coordinate values:

$$x = x + x_c, \qquad y = y + y_c$$

6. Repeat steps 3 through 5 until $x \geq y$.

Example :—

$r = 10$, for first quadrant,

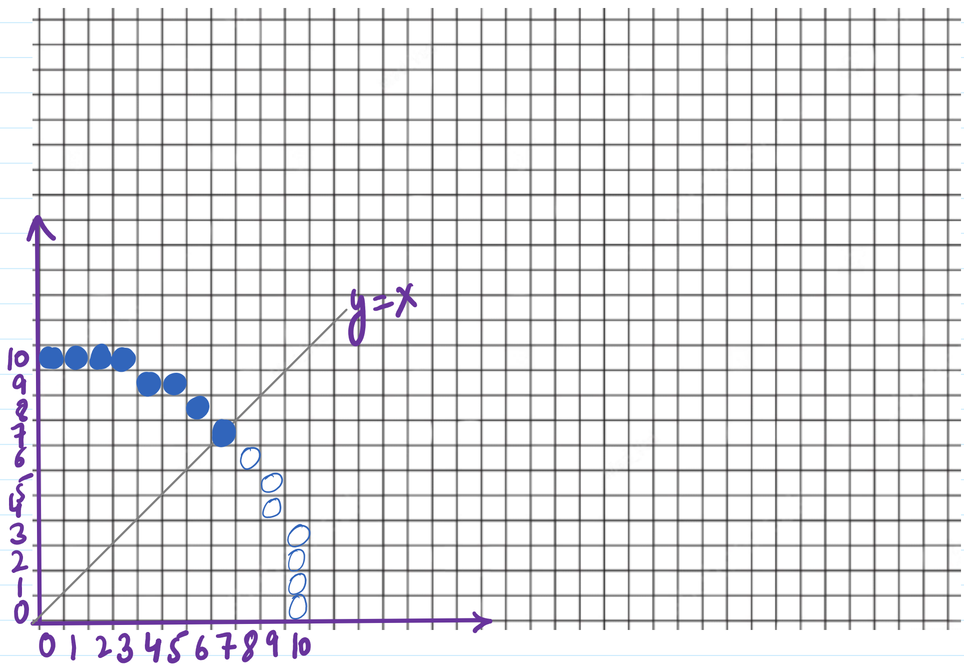
$$p_0 = 1 - r = -9$$

Origin $(0,0)$ is the center

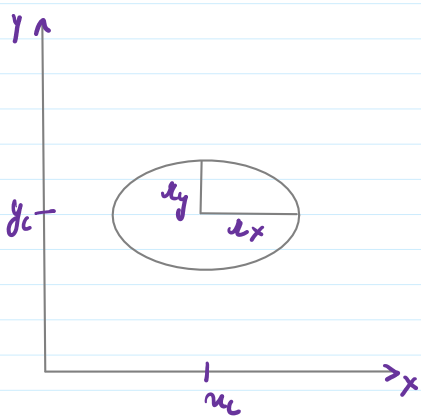
Initial point is $(x_0, y_0) = (0, 10)$ and initial incremental terms for calculating decision parameters are

$dx_0 = 0, \quad dy_0 = 20$

k	p_k	(x_{k+1}, y_{k+1})	$2x_{k+1}$	$2y_{k+1}$
0	−9	(1, 10)	2	20
1	−6	(2, 10)	4	20
2	−1	(3, 10)	6	20
3	6	(4, 9)	8	18
4	−3	(5, 9)	10	18
5	8	(6, 8)	12	16
6	5	(7, 7)	14	14



Properties of Ellipses :-



Ellipse centered at (x_c, y_c) with semimajor axis r_x and semiminor axis r_y.

$$\left(\frac{x - x_c}{r_x}\right)^2 + \left(\frac{y - y_c}{r_y}\right)^2 = 1 \qquad \text{--- (1)}$$

$$x = x_c + r_x \cos\theta \qquad \text{--- (2)}$$
$$y = y_c + r_y \sin\theta$$

"Midpoint Ellipse Algorithm"

$\Rightarrow$ Book :- Computer Graphics,

by Donald Hearn and Baker