Circle Drawing Algorithm

Circle

- A circle is defined as a set of points that are all at a given distance ‘r’ from the center position (x_c, y_c).
- The general circle equation can be written as:

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

Properties of circle

Symmetry in quadrant

- The shape of the circle is similar in each quadrat. Thus by calculating points in one quadrant we can calculate in other three quadrants.

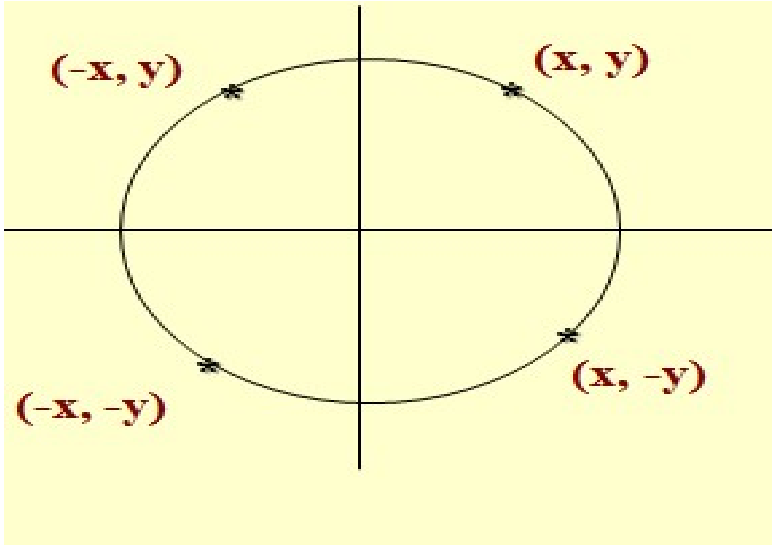


Figure: symmetry of a circle in its quadrant

Symmetry in octants

- The shape of the circle is similar in each octant. Thus by calculating points in one octant we can calculate points in other seven octants. If the point (x, y) is on the circle, then we can trivially compute seven other points on the circle.
- Therefore, we need to compute only one 45° segment to determine the circle, as shown in ***figure***.
- By taking advantage of circle symmetry in octants, we can generate all pixel positions around a circle by calculating only the points within the sector from $x = 0$ to $y = x$.

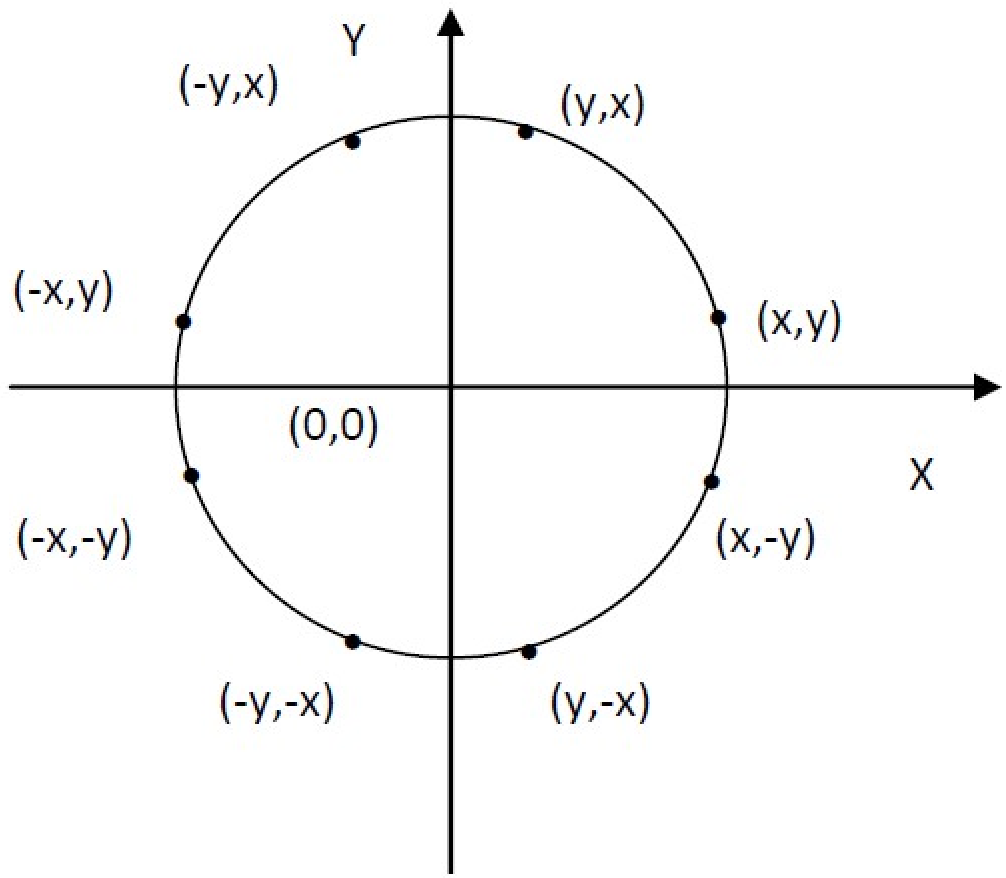


Figure: symmetry of a circle in its octant (8 way symmetry)

Methods to draw circle

- Direct method
- Trigonometric method
- Mid-point Circle method

Direct Method

- Use circle equation to calculate the positions of points on a circle circumference. Increase x-values by 1 from $x_c - r$ to $x_c + r$ and calculate associated y-value using;

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

Trigonometric Method

- Use circle equation in polar form

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

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

- To draw circle using these polar co-ordinates approach, just increment angle starting from 0 to 2π. Compute (x, y) position corresponding to increment angle.

Mid-point Circle Algorithm/Bresenham's Circle Drawing Algorithm

1. Input radius r and circle center (x_c, y_c).
2. Obtain the first point on the circumference by assuming that circle is centered on the origin i.e.,

   $(x_0, y_0) = (0, r)$.
3. Calculate the initial decision parameter as $p_0 = 5/4 - r$. (If **r is integer, then set** p_0**= 1-r.**)
4. Repeat till $x \geq y$, at each x_k position, starting at k=0, performing the following

   If $p_k < 0$

   select pixel (x_k+1, y_k)

   set $p_{k+1} = p_k + 2x_k + 3$

   else

   select pixel $(x_k+1, y_k - 1)$

   set $p_{k+1} = p_k + 2x_k - 2y_k + 5$

-Determine symmetry points in the other seven octants.

-Move each calculated pixels positions (x, y) in to circle path centered at (x_c, y_c) and plot pixels as;

1. $x = x + x_c$,
2. $y = y + y_c$

Exercise

1. ***Digitize $x^2 + y^2 = 100$ in first octant.***
2. ***Digitize a circle $(x-2)^2 + (y-3)^2 = 25$.***
3. ***Draw a circle having radius 3 units and center at (3, 2).***
4. ***Draw a circle with radius 10 units and center at origin.***
5. ***Draw a circle with radius 10 units and center at (4, -2).***

Ellipse

- An ellipse is an elongated circle.
- Therefore elliptical curves can be generated by modifying circle drawing procedures.
- The ellipse, like the circle, shows symmetry. An ellipse is symmetric in quadrants. So if one quadrant is generated then other three parts can be easily generated.
- An ellipse can be represented as:

$$\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$$

where (h, k) = ellipse center.

a = length of semi-major axis.

b = length of semi-minor axis.

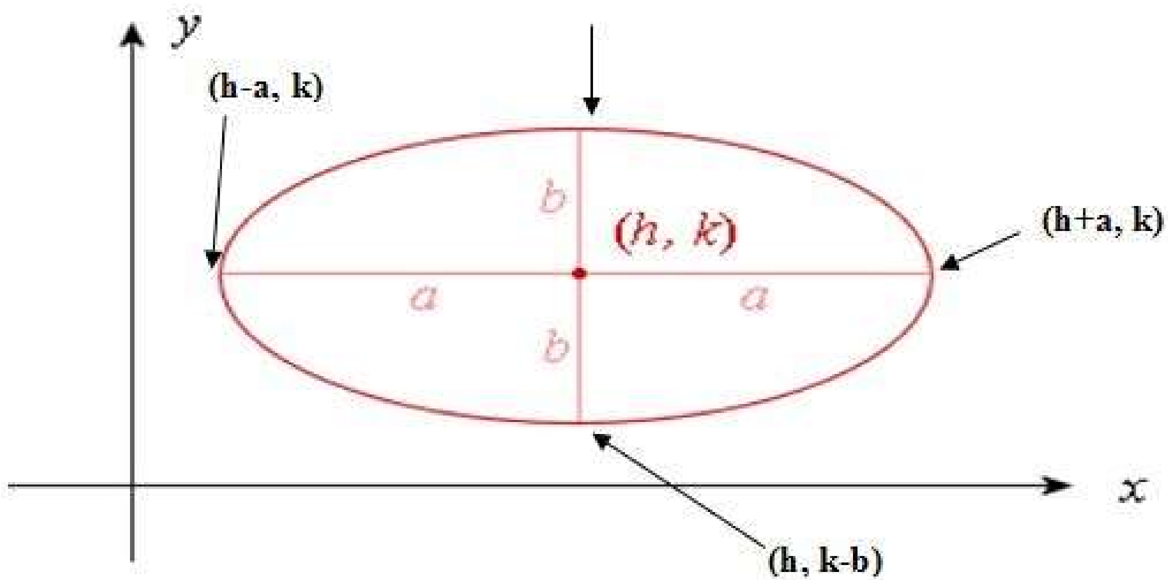


Figure: Ellipse centered at (h, k) with semi-major axis a and semi-minor axis b.

- An ellipse centered at origin (0,0) can be represented as:

  $x^2/a^2 + y^2/b^2 = 1$

Ellipse Generating Algorithms

- Direct Method
- Trigonometric Method
- Midpoint Ellipse Algorithm

Direct Method

- An ellipse can be represented as:

$$\frac{(x-h)^2}{a^2}+\frac{(y-k)^2}{b^2}=1 \qquad \textbf{.............1}$$

- Solving this equation, we get

$$\mathbf{y= k \pm} \quad b\sqrt{1-\frac{x-h^2}{a^2}} \qquad \textbf{.............2}$$

- The value of x is incremented in units of 1 from (h-a) to (h+a), and corresponding y is evaluated from equation (2).
- Plotting these computed points we can get the ellipse.

Algorithm for Direct Method

1. Input the center of ellipse (h, k), semi-major and semi-minor axis a and b respectively.
2. For each x position starting from h-a and stepping unit interval along x-direction, compute corresponding y positions as:

$$y = k \pm b\sqrt{1-\frac{x-h^2}{a^2}}$$

3. Plot the point (x, y).

Trigonometric Method

- Using polar coordinates an ellipse can be represented as;

  $x = a \cos \theta + h$

  $y = b \sin \theta + k$

  where,

  (x, y) = current coordinate

  a = length of semi-major axis

  b = length of semi-minor axis

  θ = current angle, measured in radians from 0 to 2π.

  (h, k) = ellipse center.

- In this method, θ is incremented from 0 to 2π and we compute successive values of x and y.

Algorithm for trigonometric method

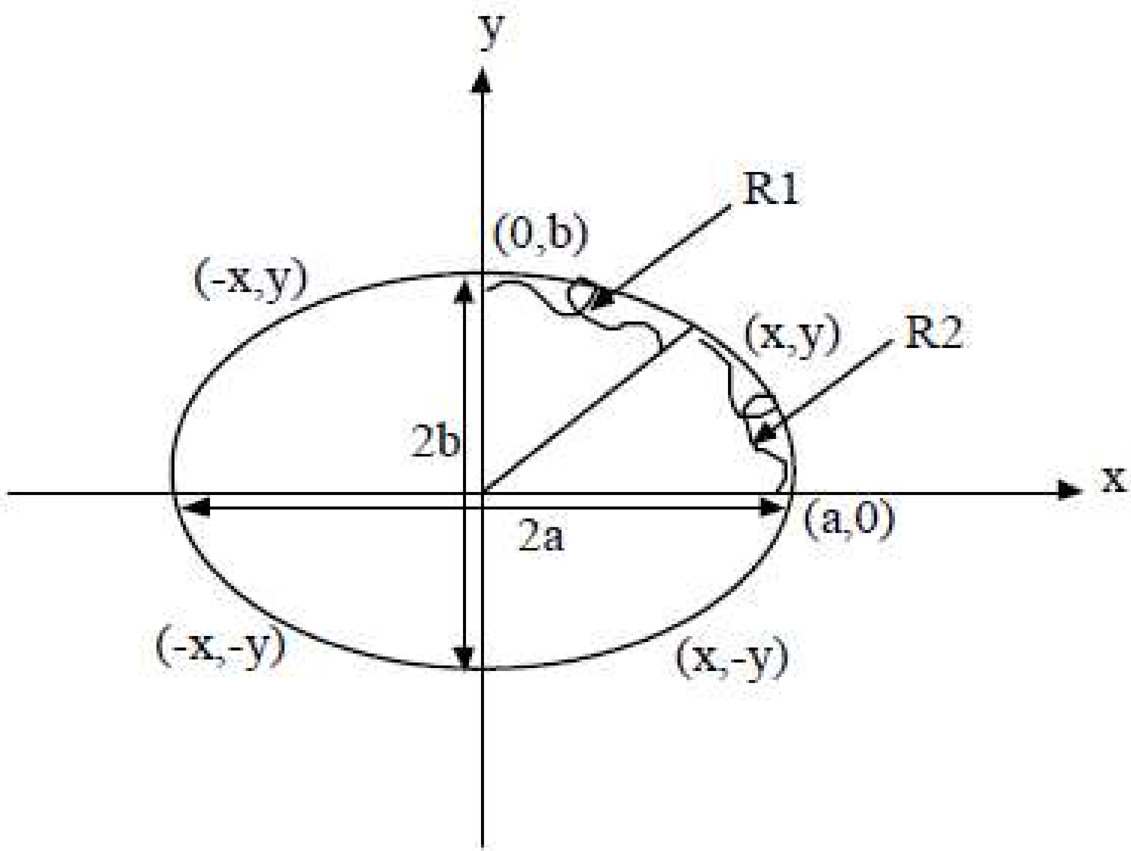
1. Input the center of ellipse (h, k), semi-major and semi-minor axis a and b respectively.
2. Starting from angle $\theta = 0$ compute boundary point of ellipse as

   $$x = a \cos \theta + h$$

   $$y = b \sin \theta + k$$

3. Plot the point(x, y).
4. Repeat until $\theta \geq 2\pi$.

Midpoint Ellipse Algorithm



- The algorithm is applied throughout the 1st quadrant according to the slope of the ellipse.
- First quadrant is divided into two parts, region 1(R1) and region 2(R2). These regions are formed by considering the slope of the curve.
- For the region (R1) where the slope of the curve is less than -1. We process by taking unit steps in x-direction and find corresponding y.
- And for the R2 where the slope is greater than -1 we take unit steps in y direction and find corresponding x. For R2, the initial point is the last calculated point in R1.

The ellipse slope is calculated from equation :

$$x^2/a^2 + y^2/b_2 = 1$$

Differentiating both sides w.r to x

$$2x/a^2 + 2y/b^2 . dy/dx = 0$$

$$dy/dx = -2b^2x/2a^2y$$

At the boundary region R1 and region R2, $dy/dx = -1$ and $2b^2x = 2a^2y$ at the boundary region.

Therefore, we move out of region 1 (R1) when $2b^2x \geq 2a^2y$.

Algorithm

1. Input center (h, k), semi-major and semi-minor axis length as a and b.
2. Obtain the first point on an ellipse by assuming that ellipse is centered on the origin i.e.

   $(x_0, y_0) = (0, b)$.
3. Compute initial decision parameter for region R1 as,

   $p1_0 = b^2 - a^2b + a^2/4$.
4. Starting at k = 0, repeat the following till $2b^2x_k < 2a^2y_k$, at each x_k position in R1, perform the following test:

   If $p1_k < 0$

   Select pixel(xk+1, yk)

   Set $p1_{k+1} = p1_k + 2b^2x_{k+1} + b^2$

   Otherwise

   Select pixel(xk+1, yk-1)

   Set $p1_{k+1} = p1_k + 2b^2x_{k+1} - 2a^2y_{k+1} + b^2$

   Determine the symmetry points in other 3 quadrants.

   Move each calculated point (x_k, y_k) on to the centered (h, k) ellipse path as

   xk = xk + h;

   yk = yk + k

5. Calculate value of the initial decision parameter at region R2 using last calculated point say (x_0, y_0) in R1 as;

   $$p2_0 = b^2(x+1/2)^2 + a^2 (y-1)^2 - a^2b^2$$

6. Repeat the following till y>0, at each x_k position in R2, starting at k = 0, perform the following test:

7. If $p2_k>0$

   Select pixel(xk, yk -1)

   Set $p2_{k+1} = p2_k - 2a^2y_{k+1} + a^2$

   Otherwise

   Select pixel(xk+1, yk-1)

   Set $p2_{k+1} = p2_k + 2a^2x_{k+1} - 2b^2y_{k+1} + a^2$

   Determine the symmetry points in other 3 quadrants.

   Move each calculated point (x_k, y_k) on to the centered (h, k) ellipse path as

   $x_k = x_k + h;$

   $y_k = y_k + k$