

Introduction

Computer contains two components.

- **Computer hardware**
Computer hardware contains the graphics workstations, graphic **input devices** and graphic **output devices**.
- **Computer Software**
Computer software contains the **Operating systems** which controls the basic operations, **software packages** used for geometric modeling like solid modeling, wire frame modeling and drafting, **Application software** which includes the program for design analysis, several application-specific software packages.

The major use of computer graphics is in design processes like engineering and architectural systems. Now almost all the products are computer designed. It is generally referred as CAD (Computer Aided Design) . It is mainly used in the design of buildings, automobiles, aircraft, computers, textiles, etc.

Survey of Computer Graphics:

- **Animations:**
 - Real time animations useful for testing performance of a vehicle or system. Wire frame displays used to watch the inner components behavior during motion.
- **Simulation:**
 - Operating a aircraft ,operating a tractor
- **Representation of graphics:**
 - Used to produce reports.(eg) bar chart, pie chart, line chart
- **Computer Art:**
 - Used in fine art, commercial art. For this a special purpose h/w is used.
- **Entertainment:**
 - Movies, Advertisement, cartoons, etc
- **Education and training:**
 - CBT computer based training.
- **Image processing:**
 - It applies the technique to modify or interpret the existing pictures.
- **Visualization:**
 - Producing graphical representation for scientific, engineering, medical data sets and processes is generally referred as scientific visualization.
- **GUI:**

- Graphical user Interface- Most of the s/w packages provides graphical interface. The major component of a graphical interface is a window manager.

Important characteristic of video display devices.

Persistence

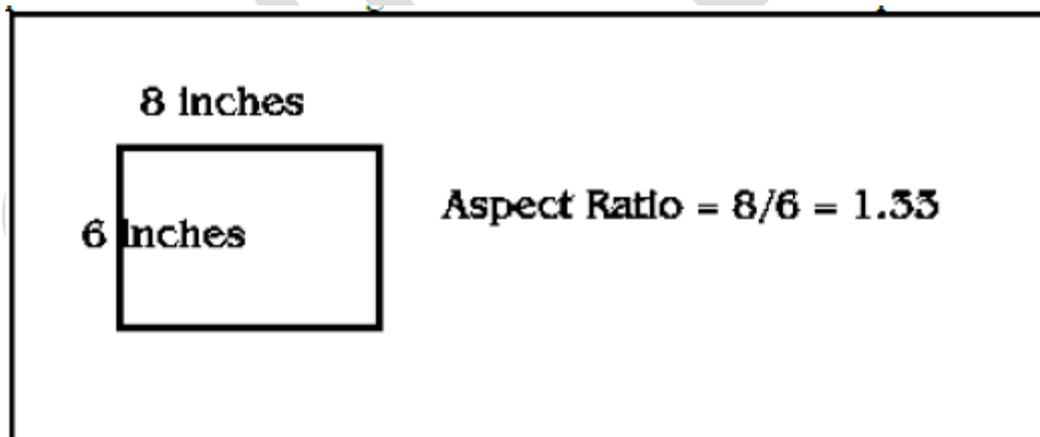
How long they continue to emit the light after the CRT beam is removed is called persistence. It is defined as the time it takes the emitted light from the screen to decompose to one-tenth of its original intensity. Lower persistence phosphors require high refresh rates to maintain a picture. The phosphor with Low Persistence used for animation. Normal persistence range is 10 to 60 micro seconds.

Resolution

The maximum No. of points that can displayed without overlap on a CRT is referred to as the resolution. It is defined as the number of points per centimeter that can be plotted horizontally and vertically

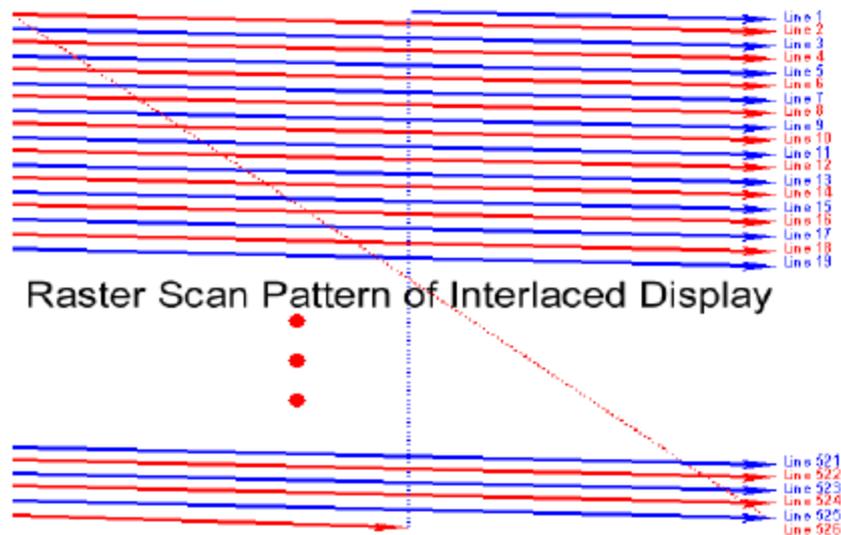
Aspect ratio

The ratio of vertical points to horizontal points necessary to produce equal length lines in both directions on the screen. AN aspect ratio of 4/5 means that a vertical line plotted with four points has the same length as a horizontal line with five points.



b. Raster Scan Displays

The most common type of graphics monitor employing a CRT is the raster scan display, based on television technology. In a raster scan system the electron beam is swept across the screen, one row at a time from top to bottom. As the electron beam is swept across the screen, one row at a time from top to bottom. As the electron beam moves across each row, the beam intensity is turned on and off to create a pattern of illuminated spots.



Picture definition is stored in a memory area called the **frame buffer** or **Refresh buffer**. This memory area holds the set of intensity values for all the screen points. Stored intensity values are then retrieved from the refresh buffer and painted on the screen one row at a time. Each screen point is referred to as a **pixel** or **pel** (Picture element).

Pixel is the smallest addressable unit in the screen. The capability of a raster scan system to store intensity information for each screen point makes it well suited for the realistic display of scenes containing suitable shading and color patterns. Home television sets and printers are examples of the other systems using raster scan methods.

High quality system uses high number of bits per pixel whereas low quality system uses less number of bits per pixel.

On a black and white system with one bit per pixel, the frame buffer is called **bitmap**. For systems with multiple bits per pixels the frame buffer is often referred to as a **Pixmap**. Refreshing on raster scan displays is carried out at the rate of 60 to 80 frames per second, although some systems are designed for higher refresh rates.

The electron beam return to the left of the screen, after refreshing each scan line is called the **horizontal retrace** of the electron beam. At the each end of the frame, the electron beam returns (**vertical retrace**) to the top left corner of the screen to the beginning of the next frame.

If the frame is displayed in single pass then that is known as **non interlaced refresh** procedures. If the frame is displayed in two passes then that is known as **interlaced refresh** procedures (In first pass all the points in the even numbered scan lines are displayed. In the second pass all the odd numbered scan lines are displayed). This is an effective technique for avoiding flicker, providing that adjacent scan lines contain similar display information.

Difference between Raster and Random

Raster scan Systems	Random scan Systems
Display controllers used to control the operation	Graphics controller or display processing unit is using to control the operation
Image is drawn using scan conversion of pixels	Using line commands, images are drawn
Selecting a particular part of the image is possible	Selecting the part of the image is not possible.
Modification on the picture is easy because of the intensity values in the frame buffer	To modify the image, it requires the redrawing
It requires high Intensity values. Because each pixel requires intensity values	It requires low intensity values
It produces the realistic image	It produces the unrealistic images
It produces jagged line drawings	It produces the smooth line drawings
High refresh rate	Low refresh rate
High resolution	High resolutions than raster scan systems

Line Drawing algorithm

Slope-intercept equation for a straight line is

$$Y=mx+b$$

m- slope of the line.

b-constant

Two end points of the line segment are (x_1, y_1) , (x_2, y_2)

Slope $m = \frac{y_2 - y_1}{x_2 - x_1} \square \Delta y / \Delta x$

Δx - x interval – $x_2 - x_1$

Δy – y interval – $y_2 - y_1$

$$\Delta y = m. \Delta x$$

$$\Delta x = \Delta y / m$$

If the interval is known, we can find the next point.

$$x_{i+1} = x_i + \Delta x \quad y_{i+1} = y_i + \Delta y / m$$

$$y_{i+1} = y_i + \Delta y \quad x_{i+1} = x_i + m \cdot \Delta x$$

we sample at unit x interval and y interval then this equation becomes

$$x_{i+1} = x_i + \Delta x \quad y_{i+1} = y_i + (1/m)$$

$$y_{i+1} = y_i + \Delta y \quad x_{i+1} = x_i + m \cdot 1$$

The above equations are for the lines which are processed from left to right.

The equations are for the lines which are processed from right to left is

$$x_{i+1} = x_i + \Delta x \quad y_{i+1} = y_i + (1/m)$$

$$y_{i+1} = y_i + \Delta y \quad x_{i+1} = x_i + m \cdot 1$$

Since m can be any real number between 0 and 1 the y value must be rounded to the nearest integer.

DDA Algorithm (Digital differential analyzer)

1. Get the two end points
2. Calculate the Horizontal and vertical difference between two end points (dx,dy)
3. The greatest difference value is taken as the length value.
4. Calculate the step value $dx = dx / \text{length}$, $dy = dy / \text{length}$
5. Start plotting the line from the first point.
6. Repeat through 9 step up to length value times
7. if $dx > dy$ and $x_a < x_b$ then increment x by 1 and increment y by m
8. if $dx > dy$ and $x_a > x_b$ then decrement x by 1 and decrement y by m
9. if $dx < dy$ and $y_a < y_b$ then increment y by 1 and increment x by 1/m
10. if $dx < dy$ and $y_a > y_b$ then decrement y by 1 and decrement x by 1/m

DDA Line Algorithm

It generates lines from their differential equations.

Advantages

1. It is the faster method for calculating pixel positions.

2. It is simplest algorithm. It does not require special skills for implementation.

Disadvantages

Floating point arithmetic in DDA algorithm is still time consuming

The algorithm is orientation dependent. Hence end point accuracy is poor.

Bresenham's Line Algorithm

This algorithm uses only integer addition, subtraction, and multiplication by 2. So it is efficient for scan converting algorithms. This algorithm increments either x or y by one unit depending on the slope of the line. The increment in the other variable is determined by examining the distance between the actual line location and the nearest pixel. This distance is called **decision variable** or the **error**.

In mathematical terms the error or decision variable is defined as

$$e = D_b - D_a$$

If $e > 0$ then the pixel above the line is closer to the true line.

Else the pixel below the line is closer to the true line.

Assume the current pixel is (x_k, y_k)

We have to find the next pixel position either $((x_{k+1}, y_k)$ and (x_{k+1}, y_{k+1})

The y coordinate on the line at the pixel position x_{k+1} is $y = m(x_{k+1}) + c$

Then the distance $d_1 = y - y_k = m(x_{k+1}) + c - y_k$

$$d_2 = (y_{k+1} - y) = y_{k+1} - m(x_{k+1}) - c$$

$$d_1 - d_2 = m(x_{k+1}) + c - y_k - y_{k+1} + m(x_{k+1}) + c$$

$$= 2m(x_{k+1}) - 2y_k + 2c - y_{k+1}$$

The error term is initially set as $e = 2\Delta y - \Delta x$ where $\Delta y = y_2 - y_1$, $\Delta x = x_2 - x_1$

Bresenham's algorithm

1. Get the two end points
2. Calculate the values $dx, dy, 2dy$ and $2dy - 2dx$ where $dx = X_2 - X_1$ and $dy = Y_2 - Y_1$
3. Calculate the starting decision parameter $d = 2dy - dx$
4. plot the first point
5. At each X_k along the line, starting at $k=0$, perform the following test

If $p_k < 0$, the next point to plot is (X_{k+1}, Y_k) and $P_{k+1} = p_k + 2dy$

Otherwise the next point to plot is (X_{k+1}, Y_{k+1}) $P_{k+1} = p_k + 2dy - 2dx$

6. Repeat step 5 for dx times.

The following program is used to for generalized bresenham algorithm, which will work for all the four coordinates.

Parallel Line algorithm

We can calculate the pixel position along the path simultaneously by partitioning the computations among the various processors available. One approach to the partitioning problem is to adapt an existing sequential algorithm to take advantage of multiple processors. Alternatively, we can look for other ways to set up the processing so that pixel positions can be calculated efficiently in parallel.

Circle Generating algorithm

Properties of the Circle

Circle is defined as a set of points that are all at a given distance r from a center position (X_c, Y_c) . This distance relationship is expressed by the Pythagorean theorem in Cartesian coordinates as

$$(X-X_c)^2 + (Y-Y_c)^2 = r^2$$

Bresenham's line algorithm for raster display is adapted to circle generation by setting up the decision parameters for finding the closest pixel for each sampling step.

A method for direct distances comparison is to test the halfway position between two pixels, to determine if this midpoint is inside or outside the circle boundary. This method is more easy. For an integer circle radius, the midpoint approach generates the same pixel position.

Midpoint Circle 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

$$P_0 = 5/4 - r$$

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

if $P_k < 0$, then 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 is (x_{k+1}, y_{k-1}) and

$$P_{k+1} = P_k + 2x_{k+1} + 1 - 2y_{k+1}$$

$$\text{where } 2x_{k+1} = 2x_k + 2, 2y_{k+1} = 2y_k - 2$$

4. Determine the symmetry points in the other seven octants

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

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

Ellipse generating Algorithm

Properties of the Ellipse

An ellipse is a set of points such that the sum of the distances from two fixed positions (foci) is the same for all points. If the distances to any two foci from any point $P=(x,y)$ on the ellipse are labeled d_1 and d_2 then the general equation of an ellipse can be stated as $d_1 + d_2$ is constant.

An ellipse in standard position is symmetric between quadrants. But it not symmetric between the two octants of the quadrant. So, we must calculate the pixel positions along the elliptical arc throughout one quadrant, then we obtained positions in the remaining three quadrants by symmetry.

Midpoint Ellipse Algorithm

1. Input r_x, r_y and ellipse center (x_c, y_c) and obtain the first point on an ellipse centered on the origin as

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

2. Calculate the initial value of the decision parameter in region 1 as

$$p_{10} = r_y^2 - r_x r_y + \frac{1}{4} r_x^2$$

3. At each x_k position in region 1, starting at $k=0$, perform the following test if $p_{1k} < 0$, the next point along the ellipse centered on $(0,0)$ is (x_{k+1}, y_k) otherwise the next point along the circle is (x_{k+1}, y_{k-1}) and

$$p_{1k+1} = p_{1k} + 2r_y x_{k+1} - 2r_y x_k + r_x$$

with

$$2r_y x_{k+1} = 2r_y x_k + 2r_y$$

$$2r_y y_{k+1} - 2r_y y_k = -2r_x$$

4. Calculate the initial value of the decision parameter in region 2 using the last point (x_0, y_0) calculated in region 1 as

$$p_{20} = r_y (x_0 + 1/2)^2 + r_x (y_0 - 1)^2 - r_x r_y$$

5. At each y_k position in region 2 starting at $k=0$, perform the following test if $p_{2k} > 0$ the next point along the ellipse centered on $(0,0)$ is (x_k, y_{k-1}) and

$$p_{2k+1} = p_{2k} - 2r_x y_{k+1} + r_x$$

otherwise the next point along the circle is (x_{k+1}, y_{k-1}) and

$$p_{2k+1} = p_{2k} + 2r_x y_{k+1} - 2r_x y_k + r_x$$

using the same incremental calculations for x and y as in region 1

6. Determine symmetry points in the other three quadrants.

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

$$X = x + x_c, Y = y + y_c$$

8. Repeat the steps for region 1 until $2r_y x \geq 2r_x y$

Attributes

Any parameter that affects the way a primitive is to be displayed referred to as an attribute parameter

Line Attributes

1. Line Type:

possible line type attribute include solid lines, dashed lines, dotted lines

2. Line Width

possible line width attribute include thicker line, thinner line and standard line.

3. Line Color

no of colors that can be provided to an output primitives depends upon the display device we used.

4. Pen and brush options

lines generated with pen or brush shapes can be displayed in various widths by changing the size of the mask. These shapes can be stored in a pixel mask that identifies the array of pixel positions that are to be set along the line path

Curve Attribute

Area Fill Attributes

Options of filling a defined region include a choice between a solid color or a patterned fill and choices for the particular colors and patterns.

Basic Fill Styles:

Hollow fill, Solid fill, Patterned fill

Character Attributes:

Text Attributes

A set of characters are affected by a particular attribute.(Font,size,style,color,alignment,height,bold,italic,)

Marker Attributes

A particular character is affected by a particular attribute(marker type, marker precision).

Antialiasing

The distortion (or) deformation of information due to low frequency sampling is called aliasing. The aliasing effect can be reduced by adjusting intensities of pixels along the line to minimize the effect of alising is called antialiasing.

We can improve the appearance of displayed raster lines by applying antialisaing methods that compensate for the under sampling process.

Methods of antialising

1. Increasing Resolution
2. Unweighted area sampling

3. Weighted area sampling
4. Thick line segments

1. Increasing Resolution

The aliasing effect can be minimized by increasing resolution of the raster display. By increasing resolution and making it twice the original one, the line passes through twice as many column of pixels and therefore has twice as many jags, but each jag is half as large in x and in y direction.

This improvement cause increase in cost of memory, bandwidth of memory and scan-conversion time. So it is a expensive method to reduce the aliasing method.

2. Unweighted area sampling

In general the line drawing algorithm select the pixels which is closer to the true line. In antialiasing instead of picking closest pixel, both pixels are high lighted. However their intensity values may differ.

In unweighted area sampling, the intensity of pixel is proportional to the amount of line area occupied by the pixel. It produces better results than does setting pixels either to full intensity or to zero intensity.

3. Weighted area sampling

In weighted area sampling small area closer to the pixel center has greater intensity than does one at a greater distance. Thus in weighted area sampling the intensity of the pixel is dependent on the line area occupied and the distance of area from the pixel's center.

4. Thick line segment

In raster displays it is possible to draw lines with thickness greater than one pixel. To produce a thick line, we have to run two line drawing algorithms in parallel to find the pixels along the line edges, and while stepping along the line we have to turn on all the pixels which lie between the boundaries.