chapter-1 imp

- history of computer graphics
1. Define computer graphics or Define. explain it's application (1+4)
2. Diff betn CAD & CAM or Describe
3. Diff betn Raster & vector display technology or Describe
4. CRT, Input & output hardware

chapter 2

- DDA or ~~Bret~~ Bresenham's Algorithm (Derive or eqn or numerical)
- circle or ellipse (explain or Derive
- 2D Translation (short notes) scaling, rotation

chapter-3

- 3D - object - projection (parallel or prospective) diff
- 3D Transformation
- ~~3D~~ polygon table, polygon surface
- Hidden line & surface detection technique (any one with ex. of a figure)
- lightning model
- shading model

chapter 4

- Animation sequence (explain or derive)
- morphing technology

chapter-5

- ~~phases~~ features of Graphic package design
- issues in Graphic design
- Graphic file format, Anti aliasing

chapter-6

-

Computer Graphics :-

- Computer Graphics is a field related to the generation of graphics using computer. It includes the creation, storage and manipulation of images of objects.
- These objects comes from diverse field such as medicine, physical, mathematical, engineering, architecture, entertainment, advertisment etc.
- It is related to the generation & the representation of graphics by a computer using specialized graphics hardware & software. The graphics can be photographs, drawings, movies or simulation etc.
- Raster : A rectangular array of points & dots. (represented on integer coordinate)
- pixel : one dot or picture element of the raster.
- Scan line: A row of pixel
- Bit map: ones & zeros representation of rectangular array of point on the screen.

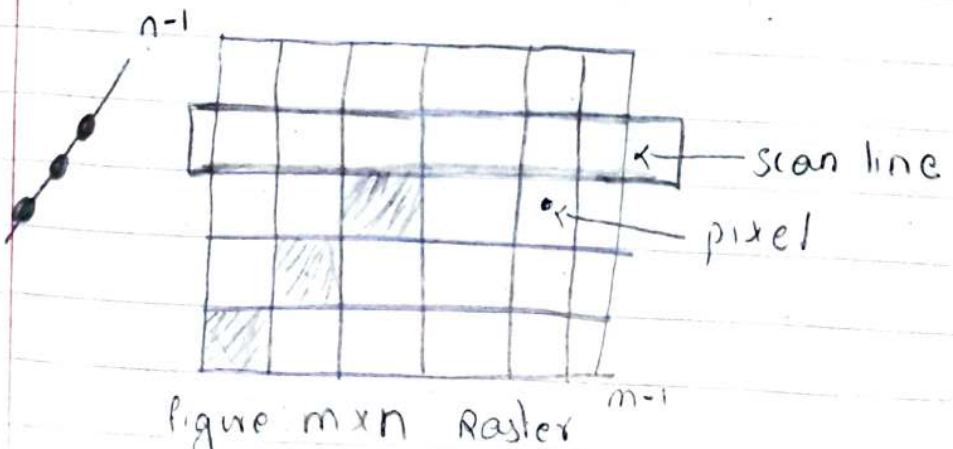


figure m x n Raster

Jul

Application of CG

- Entertainment
- Education & training
- Computer Animation
- Image processing
- Research
- Computer Art
- Cartography
- Simulation & modeling
- Graphical user interface (GUI)
- Computer Aided Design (CAD)
- Computer Aided Manufacturing (CAM)

* Entertainment

Computer graphics method are now commonly used in making motion pictures, music video, & tv shows.

- Images are drawn in wire-frame and will be shaded with rendering methods to produce solid surfaces.

- CG allows the user to make animated movies, cartoon & games.

- CG are also used to introduce virtual characters to movies like characters in "Lord of the Rings."

- CG is used to create scenes. It is used for special effects & animation, cartoons etc.

* Education & training.

-) The process of gaining knowledge, skill and development from study or training is known as education.

-> The process of learning the skills one needs to do a particular job or activity is known as Training.

- Training refers to an act of inculcating specific skills in a person. Education is all about gaining theoretical knowledge in the classroom or any institution.

- Training is a way to develop specific skills, whereas education is a typical system of learning.

computer Animation

- The process used for digitally generating animated images is known as computer animation.

- Animation is a good major for many undergraduate students.

- computer animation is a general term for a kind of visual digital display technology that simulates

moving objects on-screen.

Image processing

-> Digital image processing is the use of a digital computer to process digital images through an algorithm.

- A method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it is Image processing.

- It is used to find out various patterns and aspects in images.

Research

- The systematic investigation into and study of materials and sources in order to establish facts and reach new conclusions.

- It plays an important role in discovering new treatments, and making sure that we use existing treatments in the best possible ways.

computer Art

→ Computer Art, manipulation of computer-generated images (pictures, designs, scenery, portraits, etc). as part of a purposeful creative process.

→ computer Art also called Digital Art.
There are different types of Digital Art. They are:

- Data-moshing
- Dynamic painting
  - 2D computer Graphics
  - 3D computer Graphics
    - pixel Art
    - Digital photography
    - photo-painting

cartography

→ The study and practice of making maps is known as cartography.

- It frequently use digital cameras and scanners in addition to satellite images to capture visuals for a map.

simulation & modeling

- modeling and simulation is the use of models (eg: physical, mathematical, or logical representation of a system, entity, phenomenon, or process) as a basis for simulations to develop data utilized for managerial or technical decision making.

- simulation modeling solves real-world problems safely and efficiently.

Graphical user interface (GUI)

→ The Graphical user Interface is a form of user interface that allows users to interact with electronic devices through graphical icons and audio indicator such as primary notation, instead of text-based user interfaces, typed command labels or text navigation.

- The actions in a GUI are usually performed through direct manipulation of the graphical elements.

Computer Aided Design (CAD)

→ Computer Aided design is a way to digitally create 2D drawings and 3D models of real-world products.

- examples: AutoCAD, 3ds Max, ~~and~~ etc.

- CAD Engineers help design and develop 2D plans and 3D models necessary in product Engineering.

CAD (Computer Aided Design)

- CAD system generate accurately, scale mathematical models based on user / input.

- Individual models are then integrated as components of an assembly to create a final product through which exact fit of the parts can be checked.

computer + Designing software = CAD

- computer technology used for mainly product designing

Advantages of CAD

- Minimizes the requirement of huge numbers of an expensive draftperson in designing of a product.

→ It can be used directly in order to generate cutting data for CNC machines. (computerrized Numerical control).

→ Scaling, re-scaling modification in drawing & models is easier & automatic and accurate.

- storage & retrieval of models is easier
- Design data can be shared in computer Aed manufacturing management system.

- precise 3D models can be examined before making

expensive materials.
- It increases the speed of production & requires less labour.

- multiple copies can be stored, printed & shared electronically, which eliminates the need for storing large paper drawings.

Disadvantages of CAD

- power cuts & viruses can be problematic for the computerized system.

→ industrial version of the software could be very expensive to buy especially for startup costs.

→ Traditional drafting skills will be lost as they become unnecessary.

- Expensive training would be required to use the software, which can be time consuming & costly.

* CAM (Computer Aided Manufacturing)

- It is computer technology used for mainly manufacturing purpose.
- It is evolving a central element in many productions.
- It include a ~~broo~~ broad range of procese to be carried out automatically such as cutting, turning, milling, routing & even printing of solid materials.
- After designing & analysing a product, it is manufactured where computers are involved in manufacturing like checking whether the product can be made or made by which process, how much time it is going to take etc.
- in simple word, computer systems used in planning, managing & controlling the operation of manufacturing plant is called CAM.

computer + Manufacturing Tools = CAM

* Advantages of CAM

- Faster than traditional machines and tools used by hand.
- → More accurate than hand manufacturing methods
- → Repeatable manufacturing for large quantities

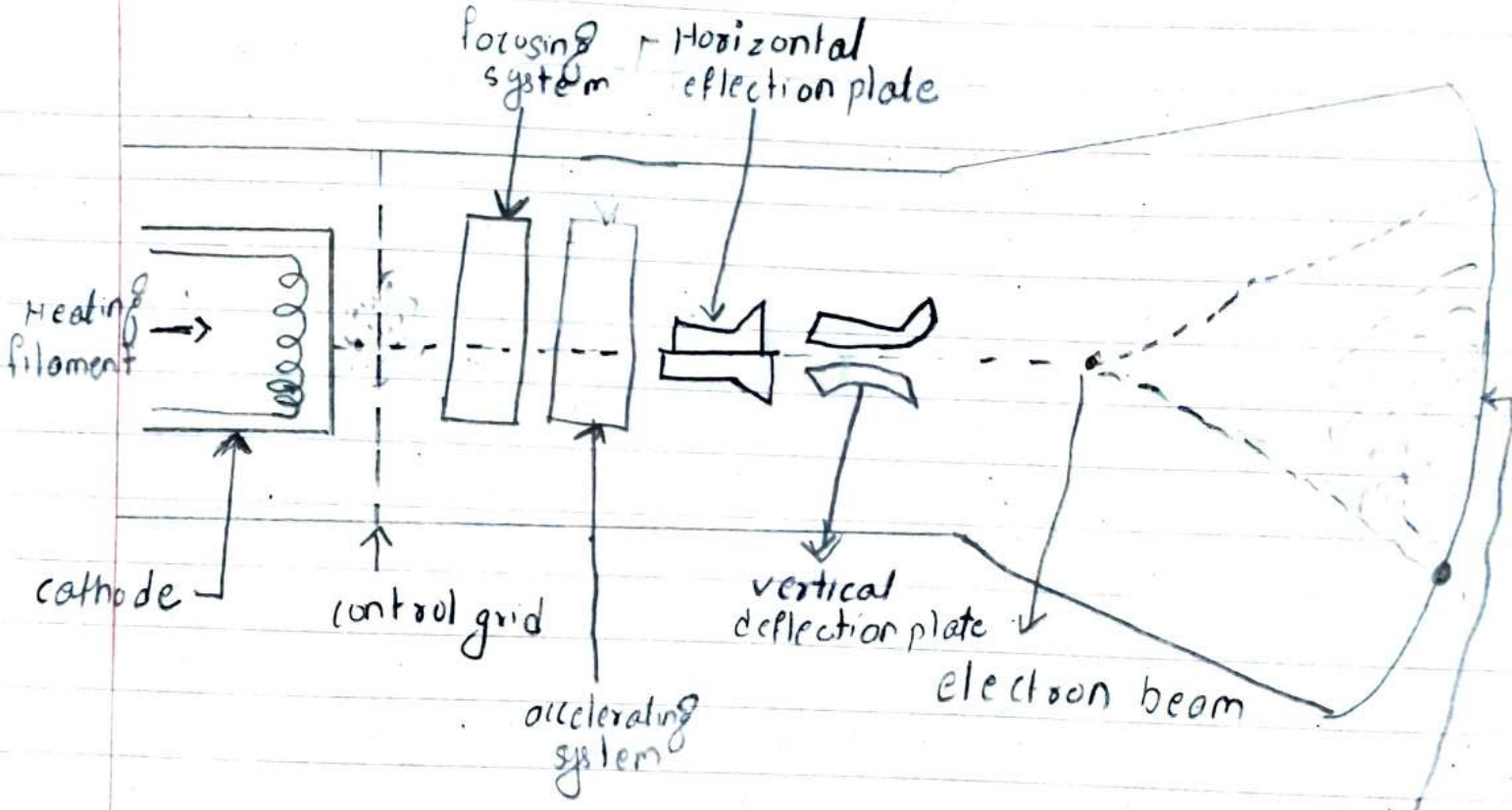
- machines can run 24/7 with little human interaction.

- can produce work directly from CAD files

Disadvantages

- expensive set up costs for equipment.
- Training costs and times.
- CAM machines need specialist engineers when they need maintaining or reparing.
- CAM machines can do work that is traditionally done by skilled workers - leading to redundancy and unemployment in the manufacturing industry.

CRT (Cathode Ray Tube)



~~pre~~ persistance :-

- How long a phosphor continous to emit light after the electron ~~beames~~ beam is removed.

- persistance of phosphor is defined as time it take for emitted light to decay to 1/10 (10 %) of it's original intensity. Range of persistence of different phosphor can react many seconds.

- The phosphor used for graphics display device usually have persistence of 10 to 60 microsecond.

Resolution:
Resolution is defined as the maximum number of points that can be displayed horizontally & vertically without overlap on display devices.

Horizontal scan rate
- The horizontal scan rate is the number of scan lines per second. The rate is approximately the product of the refresh rate and the number of scan lines.

Aspect ratio
- It gives the ratio of vertical point to horizontal point necessary to product equal length lines in both directions on the screen.

- As aspect ratio of 3/4 means that a vertical line plotted with 3 points has the equal/same length as a horizontal line plotted with 4 points.

Refresh Rate
- light emitted by phosphor fades very rapidly. So to keep the drawn picture glowing constantly, it is required to re draw the picture repeatedly & quickly directing the electron beam back

over the same point. The no. of times/sec the image is redrawn to give a feeling of non-flickering pictures is called refresh rate.

- If Refresh rate decreases, flickers develops.
- Refresh rate above which flickering stops & steady it may be called as CFF (Critical fusion frequency).

Display Technology
↳ Raster scan display
↳ Random/vector scan display

(1) ↳ Raster scan display

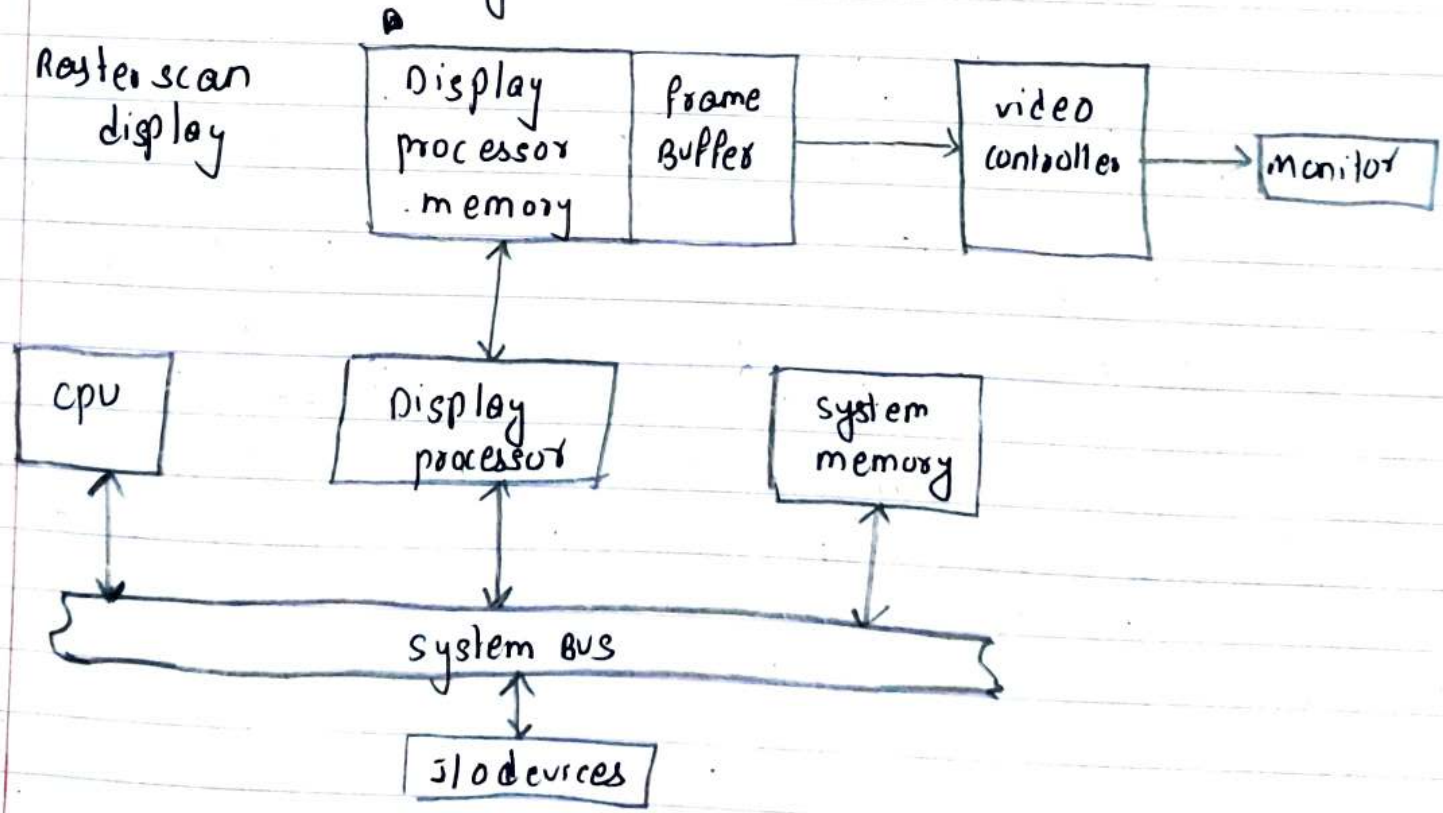
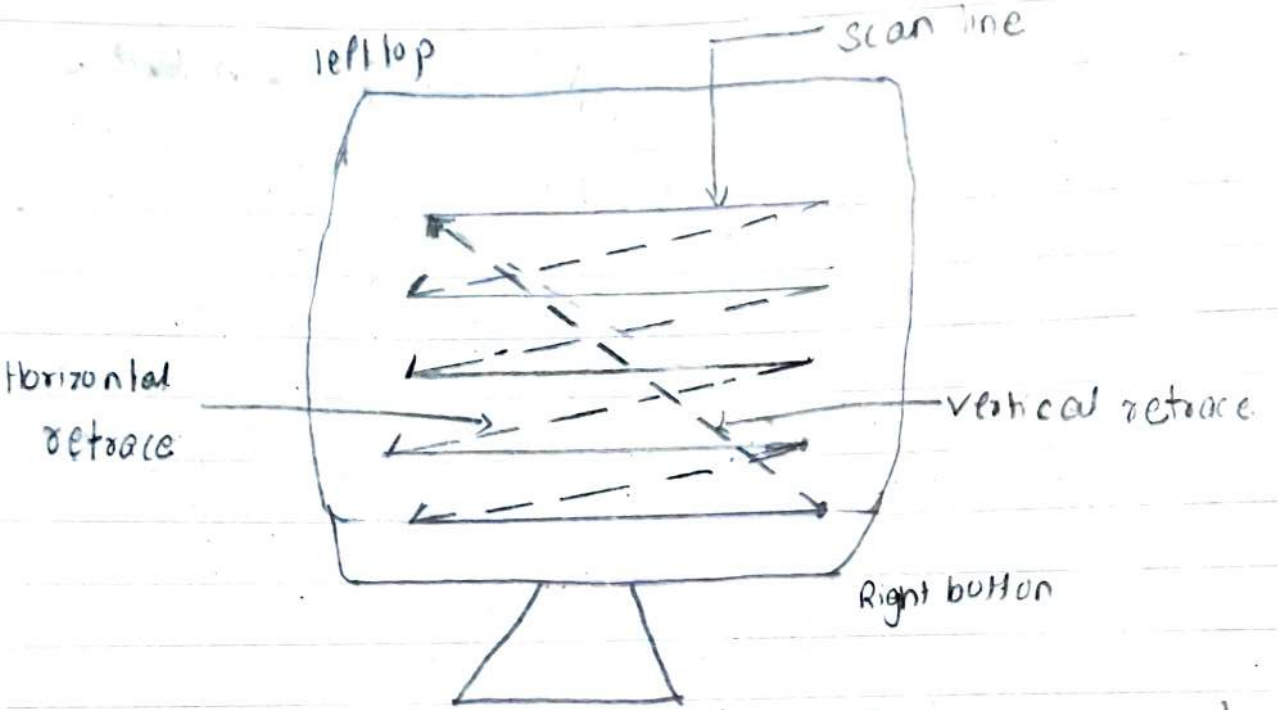


figure: Architecture of Raster-graphic system.



scanning process of Raster scan display

Raster & vector Graphics

A Raster images is made up of pixels, each a different colour, arranged to display an image where as a vector image is made up of paths, each with a mathematical formula (vector formula), that tells the path how it is shaped & what colour it is bordered with or ~~paint~~ filled by.

	Raster graphics	Random/ vector graphics
1.	Raster graphics are composed of pixels.	Vector graphics are composed by paths.
2.	Raster image pixels don't keep on their appearance as size increases, when you blow up photograph up, it becomes blurry for this reason.	vector images keep on appearance regardless of size, since the mathematical formula's dictate how the image is rendered.

↳ Raster scan display

↳ In Raster scan system the electron beam is swept across the screen, one row at a time from top to bottom. As electrons beams moves across each row the beam intensity is turned on and off to create a pattern of illuminated spots.

↳ pictures definitions is stored in memory called frame buffer or refresh buffer. This memory hold the set of intensity values for all the screen points stored intensity values are then retrieved from the frame buffer and painted on the screen one row at a time.

↳ Returning of electron beam from right end to left end after refreshing each sun line is called horizontal retrace.

- At the end of each frame, the electron beam returns to the top-left corner to begin next frame called vertical retrace.

- There is a special purpose processor called video controller or display controller, is used to control the operation of the display device.

- When a particular command is called by the application program, the graphics subroutine package sets the appropriate pixels in the frame buffer. The video controller then cycles through the frame buffer, one scan line at a time. It will bring a value of each pixel contained in the frame buffer & uses it to control the intensity of the CRT electron beam.

- The display processor is a seperate processor that performs graphics function such as scan conversion & raster operation.

- System memory holds data & those program that execute on the CPU.

- The display processor memory holds data plus program that perform scan conversion & raster operation.

- The frame buffer stores displayable image created by scan conversion & raster operation.

Random scan (vector) Display

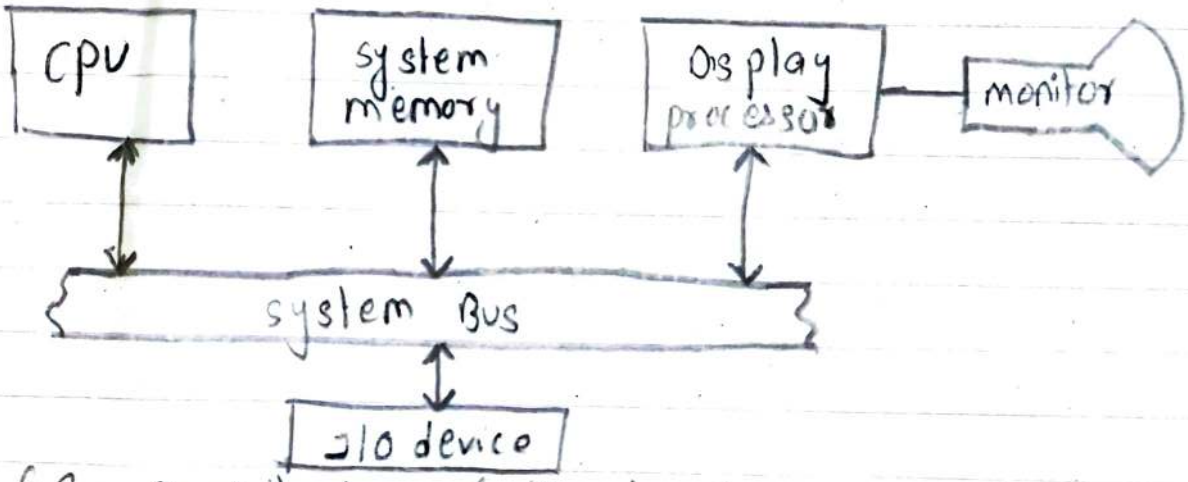


fig: Architecture of Random/vector display system

- In random scan system, the electron beam is directed only to the part of screen where the picture is to be drawn. It draws a picture one line at a time, so it is also called vector display.
- picture definition is stored as a set of line drawing commands in an area of memory called refresh display file.

- To display a picture, the system cycles through the set of commands is the display file. After all commands are processed, the system cycle backs to the first line command in the list.

- The graphics commands in the application program are translated by the graphics package into a display list (display files) stored in system memory. The display list is accessed by the display processor to refresh the screen. The display processor cycles through each command in the display list once during each refresh cycle.

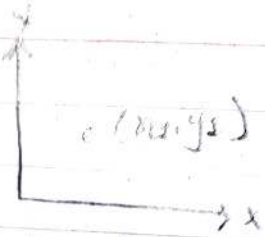
- Graphics are drawn on a vector display system by directing the electron beam along components lines of the pictures.

Unit-2

2- Dimension Graphics

Scan conversion:

It is the process used to digitize or Rasterize pixel data available on frame buffer (memory).



- The process of representing continous ~~data~~ graphics object as a collection of discrete pixel is called scan conversion.

  eg: line is defined by it's two end points & the line equation.

Scan conversion of point: (x, y)

- scan conversion of a point requires the two data are pixel position & color for display.

- In c, a point be scan converted using function putpixel() defined in library.
  Header file is <graphics.h>

putpixel(x, y, color), here x & y represent pixel position on 2D Display.

scan conversion of line:

equation of straight line = $y = mx + c$

where,

m = slope of line

$m = y/x$

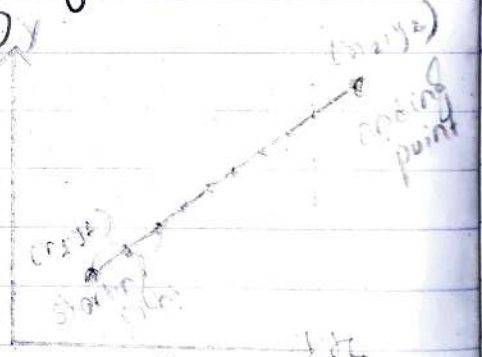
$m = \Delta y / \Delta x$

Line Drawing Algorithm

↳ DDA (Digital Differential Algorithm) / (Increment Algorithm)
↳ Bresenham's line Drawing algorithm

1) DDA (Digital Differential Algorithm)

It is scan conversion line algorithm for drawing a line base on calculating either Δx & Δy using equation $y = mx + c$ — ①



$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{\Delta y}{\Delta x} \quad — ②$$

for any interval of Δx, line eqⁿ for left to right

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

case I :
If $m < 1$
If : $|m| < 1$, $\Delta x = 1$
Then $x_{k+1} = x_k + 1$ — ③
$\Delta y = m$ Then,
$\Delta y = m$
$y_{k+1} = y_k + m$ — ④

case II
If $|m| > 1$, $\Delta y = 1$
$y_{k+1} = y_k + 1$
Then,
$\Delta x = 1/m$
$x_{k+1} = x_k + 1/m$

case III
If $m = 1$
$y_{k+1} = y_k + 1$
$x_{k+1} = x_k + 1$

In general,

$$x_{k+1} = x_k \pm 1$$
$$y_{k+1} = y_k \pm 1 \quad \text{for } m < 1$$

$$x_{k+1} = x_k + 1/m$$
$$y_{k+1} = y_k + 1 \quad \text{for } m > 1$$

Examples

Q1. Digitized the line with end points (0,0) & (4,5) using DDA.

Soln,

$(x_1, y_1) = (0,0)$

$(x_2, y_2) = (5,4)$

we have,

slope $(m) = \frac{y_2 - y_1}{x_2 - x_1} = \frac{4-0}{5-0} = 0.8$

$m = 0.8$ which is less than 1 i.e. $m < 1$

For DDA algorithm

$$x_{k+1} = x_k + 1$$
$$y_{k+1} = y_k + m$$

x	y	x-plot	y-plot	(x,y)
0	0	0+1 = 1	0+0.8 = 0.8	(1,1)
1	0.8	1+1 = 2	0.8+0.8 = 1.6	(2,2)
2	1.6	2+1 = 3	1.6+0.8 = 2.4	(3,2)
3	2.4	3+1 = 4	2.4+0.8 = 3.2	(4,3)
4	3.2	4+1 = 5	3.2+0.8 = 4.0	(5,4)

① Digitized the line with end points (0,0) & (20,10)

② (3,7) & (8,3)

③ (20,10) & (30,18)

④ (6,12) & (10,5)

① (0,0), (20,10)

soln,

$(x_1, y_1) = (0,0)$

$(x_2, y_2) = (20,10)$

we have,

$$\text{slope } (m) = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= \frac{10-0}{20-0}$$

$$= 0.5$$

$m = 0.5$ which is less than 1 i.e. $m < 1$

for DDA algorithm,

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

$y_{k+1} = y_k + m$

x	y	x-plot	y-plot	(x,y)
0	0	0+1=1	0+0.5=0.5	(1,1)
1	0.5	1+1=2	0.5+0.5=1	(2,1)
2	1	2+1=3	1+0.5=1.5	(3,2)
3	1.5	3+1=4	1.5+0.5=2	(4,2)
4	2	4+1=5	2+0.5=2.5	(5,3)
5	2.5	5+1=6	2.5+0.5=3	(6,3)
6	3	6+1=7	3+0.5=3.5	(7,4)

② ~~(3,7) & (8,3)~~

7	3.5	7+1=8	3.5+0.5=4	(8, 4)
8	4	8+1=9	4+0.5=4.5	(9, 5)
9	4.5	9+1=10	4.5+0.5=5	(10, 5)
10	5	10+1=11	5+0.5=5.5	(11, 6)
11	5.5	11+1=12	5.5+0.5=6	(12, 6)
12	6	12+1=13	6+0.5=6.5	(13, 7)
13	6.5	13+1=14	6.5+0.5=7	(14, 7)
14	7	14+1=15	7+0.5=7.5	(15, 8)
15	7.5	15+1=16	7.5+0.5=8	(16, 8)
16	8	16+1=17	8+0.5=8.5	(17, 9)
17	8.5	17+1=18	8.5+0.5=9	(18, 9)
18	9	18+1=19	9+0.5=9.5	(19, 10)
19	9.5	19+1=20	9.5+0.5=10	(20, 10)

H-W to C-W

(11) Digitized the line with and end points (3,7) & (8,3)

⇒ Soln,

$(x_1, y_1) = (3, 7)$

$(x_2, y_2) = (8, 3)$

We have,

$$m = \frac{\Delta y}{\Delta x} = \frac{3-7}{8-3} = \frac{-4}{5} = -0.8$$

which is less than 1.

$\therefore m < 1$ in DDA.

we have,

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

$y_{k+1} = y_k + 1$

X	Y	X-plot	Y-plot	(x, y)
3	7	3+1=4	7 - 0.8 = 6.2	(4, 6)
4	6.2	4+1=5	6.2-0.8 = 5.4	(5, 5)
5	5.4	5+1=6	5.4-0.8 = 4.6	(6, 5)
6	4.6	6+1=7	4.6 -0.8 = 3.8	(7, 4)
7	3.8	7+1=8	3.8-0.8 = 3	(8, 3)

DDA Algorithm

Step 1: Declare the variables, x_1, y_1 and x_2, y_2, dx, dy delx and dely as real and step(k) as integer no.

Step 2: perform $dx = x_2 - x_1$ & $dy = y_2 - y_1$

Step 3: Test if $|dy| < |dx|$ then
step (k) = |dx|
else step (k) = |dy|

step 4: set delx = dx / step (k)
dely = dy / step (k)
$x = x_1$, $y = y_1$

Step 5: plot (x, y)

step 6: Do for k = 1 to step (k)
x = x + delx
y = y + delx
plot (x, y)

step 7 = End do.

Digitized the line with end points (0,0) & (4,5) using DDA Algorithm.

Soln,

Given,

$(x_1, y_1) = (0,4)$

$(x_2, y_2) = (4,5)$

According to algorithm

$dx = x_2 - x_1 = 4 - 0 = 4$

$dy = y_2 - y_1 = 5 - 0 = 5$

since,

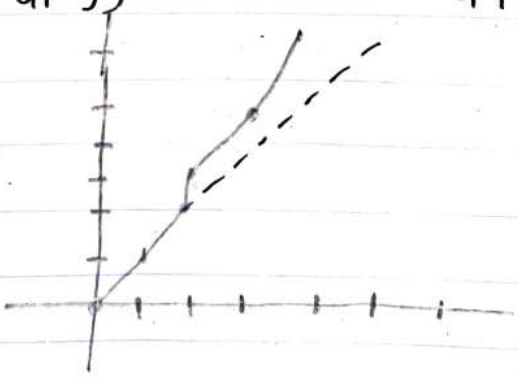
$dy > dx$, $k = dy = 5$

we have,

$delx = dx/k = 4/5 = 0.8$

$dely = dy/k = 5/5 = 1$

Iteration (k)	plot [round(x), round(y)]	x+Dx	y+Dy
0	(0,0)	0+0.8 = 0.8	0+1 = 1
1	(1,1)	0.8+0.8 = 1.6	1+1 = 2
2	(2,2)	1.6+0.8 = 2.4	2+1 = 3
3	(2,3)	2.4+0.8 = 3.2	3+1 = 4
4	(3,4)	3.2+0.8 = 4	4+1 = 5
5	(4,5)	4+0.8 = 4.8	5+1 = 6



Advantages of DDA

- It is faster method then direct use of line equation, $y = mx + c$.
- It is simple to understand because each step involves just two additions.
- It requires no special skills for implementation.
- This method doesn't use multiplication theorem.
- It allows us to detect the changes in the value of x & y so, plotting of same point twice is impossible.

Disadvantages of DDA

- Slope is stored in ~~plotting~~ floating point number.
- Accumulater of round off error in successive addition can cause pixel position to drift away from the actual line path for long line segment.
- Rounding operation and floating point arithmetic are time consuming.
- It is more suitable for generating line using the software but it is less suited for hardware implementation.

Bresenham's Algorithm :

→ DDA includes calculation related to m and 1/m which is little complicated. Bresenham's ~~improves~~ improves DDA algorithm by only involving integer calculations.

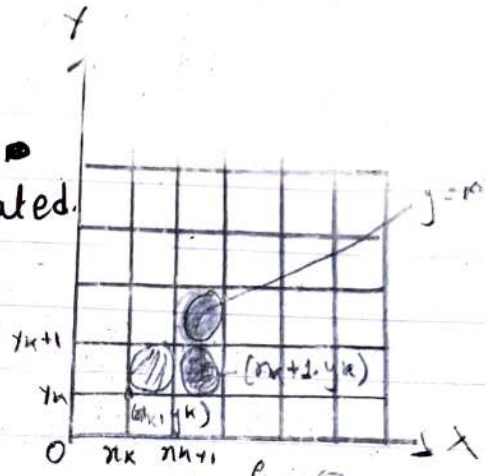


fig (1)

- BLA selects the best pixel co-ordinates by testing the sign of an Integer parameter whose value is proportional to the difference b/n the seperation of two pixel actual line path.

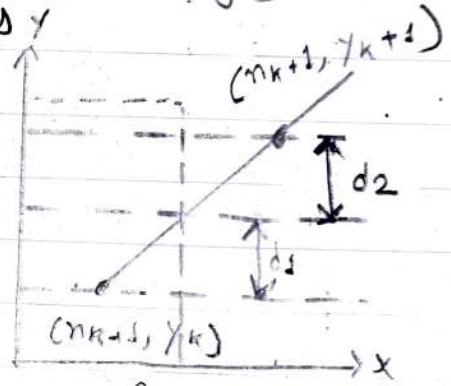


fig (2)

case I: $m < 1$ or $m > 0$.

- Let (x_k, y_k) is pixel determined at k^{th} step, than the next pixel is to be plotted may be either (x_k+1, y_k) or (x_{k+1}, y_{k+1}).
- Let d_1 & d_2 be the seperation of pixel position (x_k+1, y_k) & (x_k+1, y_k+1) from the actual line line path ~~by~~ $y = mx + c$
~~From the figure (2) above.~~
Then, at sampling position (x_k+1),

$$Y = m(x_{k+1}) + c \quad \text{—— (1)}$$

from the figure (2) above,

$$d_2 = y_{k+1} - y$$

$= y_{k+1} - m(x_k + 1) - c$ —— (2)

And

$d_1 = y - y_k$

$= m(x_k + 1) + c - y_k$ —— (3)

- The difference between these two seperation is given by equation (2) and (3) difference

$d_1 - d_2 = 2m(x_k + 1) - (y_k + 1) - y_k + 2c$

$= 2m(x_k + 1) - 2y_k + 2c - 1$ —— (4)

let us define a decision parameter p_k for the k^{th} step in the line algorithm by

~~$p_k = \Delta x(d_1 - d_2)$~~

$p_k = \Delta x(d_1 - d_2)$

since, $\Delta x > 0$, Therefore,

$p_k < 0$, if $d_1 < d_2$

$p_k > 0$, if $d_1 \geq d_2$

$\therefore p_k = \Delta x(d_1 - d_2) = (2m(x_k + 1) - 2y_k + 2c - 1)$

$= \Delta x \left[2\frac{\Delta y}{\Delta x}(x_k + 1) - 2y_k + 2c - 1 \right]$

$= \frac{2 \times \Delta x \cdot \Delta y}{\Delta x} = \Delta y$

$= \Delta x \cdot 2\frac{\Delta y x_k}{\Delta x} + \frac{\Delta x 2\Delta y}{\Delta x} - 2\Delta x y_k + 2\Delta x c - \Delta x$

$= 2\Delta y x_k + 2\Delta y - 2\Delta x y_k + 2\Delta x c - \Delta x$

$= 2\Delta y x_k - 2\Delta x y_k + 2\Delta y + 2\Delta x c - \Delta x$

$= 2\Delta y x_k - 2\Delta x y_k + 2\Delta y + \Delta x (2b-1)$

$p_k = 2\Delta y \cdot x_k - 2\Delta x y_k + b$ — (5)

where the const. $b = 2\Delta y + \Delta x (2b-1)$

now,

for next step,

$\therefore p_{k+1} = 2\Delta y \cdot x_{k+1} - 2\Delta x y_{k+1} + b$ — (6)

from eqn (5) & (6)

$p_{k+1} - p_k = 2\Delta y (x_{k+1} - x_k) - 2\Delta x (y_{k+1} - y_k)$

$p_{k+1} = p_k + 2\Delta y (x_{k+1} - x_k) - 2\Delta x (y_{k+1} - y_k)$

$\therefore p_{k+1} = p_k + 2\Delta y - 2\Delta x (y_{k+1} - y_k)$

where,

$y_{k+1} - y_k = 0$ or 1

if $p_k \geq 0$, then we plot upper pixel

Therefore, $y_{k+1} = y_k + 1$

$\therefore p_{k+1} = p_k + 2\Delta y - 2\Delta x$ — (7)

if $p_k < 0$, then we plot lower pixel

$y_{k+1} = y_k$

$\therefore p_{k+1} = p_k + 2\Delta y$ — (8)

Therefore, initial decision parameter p_0 is given by

$p_0 = 2\Delta y x_0 - 2\Delta x y_0 + b$ [:- from eqn (5)]

$= 2\Delta y x_0 - 2\Delta x y_0 + 2\Delta y + \Delta x (2c-1)$

$= 2\Delta y x_0 - 2\Delta x y_0 + 2\Delta y + 2\Delta x c - \Delta x$

$= 2\Delta y x_0 - 2\Delta x y_0 + 2\Delta y + 2\Delta x (y_0 - \frac{\Delta y}{\Delta x} x_0) - \Delta x$

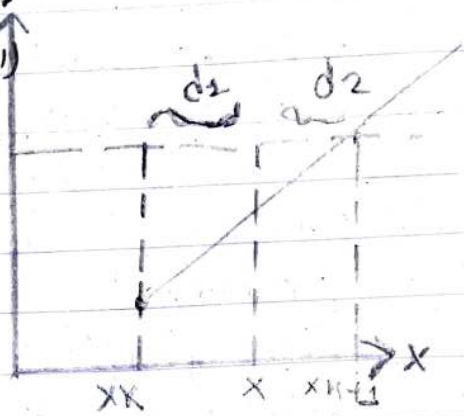
$= 2\Delta y x_0 - 2\Delta x y_0 + 2\Delta y + 2\Delta x y_0 - 2\Delta y x_0 - \Delta x$

$\therefore p_0 = 2\Delta y - \Delta x$

Assignment - I

Case II : $m > 1$

- let (x_k, y_k) is pixel determined at k^{th} step. Then the next pixel to be plotted may be either (x_{k+1}, y_k) or (x_{k+1}, y_{k+1})



- let d_1 & d_2 be the seperation of pixelposition (y_{k+1}, n_k) & (y_{k+1}, x_{k+1}) from the actual line path by $y = mn + c$

Then at sampling position (y_{k+1}),

$$\Rightarrow x = \frac{y - c}{m} = \frac{y_{k+1} - c}{m} \quad \text{---(1)}$$

From the figure (2) above,

$$d_2 = x_{k+1} - x$$
$$= x_{k+1} - \frac{(y_{k+1}) - c}{m} \quad \text{---(2)}$$

And $$d_1 = x - x_k$$
$$= \frac{y_{k+1} - c - mx_k}{m} \quad \text{---(3)}$$

- The difference between these two seperation is given by equation (2) and (3) differen ce

$$d_1 - d_2 = \frac{y_{k+1} - c}{m} - x_k - \left(x_{k+1} - \frac{(y_{k+1}) - c}{m} \right)$$

$$= \frac{y_{k+1} - c - mx_k - m(x_{k+1}) + y_{k+1} + c}{m}$$

$$\frac{y_{k+1} - C - mx_k - mx_k + m + y_{k+1} + C}{m}$$

$$= \frac{2y_{k+1} - 2mx_k - m}{m}$$

$=$

$$d_1 - d_2 = \frac{y_{k+1} - C - mx_k}{m} - \left(x_{k+1} - \left(\frac{y_{k+1}}{m} \right) - C \right)$$

$$= \frac{y_{k+1} - C - mx_k}{m} - x_{k+1} + \frac{y_{k+1}}{m} + C \quad \text{— (4)}$$

let us define a decision parameter p_k for the k^{th} step in the line algorithm by

$$p_k = \Delta x (d_1 - d_2)$$

since, $\because \Delta y > 0$, Therefore,

$p_k < 0$, if $d_1 < d_2$

~~$p_k < 0$~~ $p_k > 0$, if $d_1 > d_2$

$$\therefore p_k = \Delta y (d_1 - d_2) = \left(\frac{y_{k+1} - C - mx_k}{m} - x_{k+1} + \frac{y_{k+1} + C}{m} \right)$$

$$= \Delta y \left[(y_{k+1} - C) \times \frac{\Delta x}{\Delta y} - mx_k + x_{k+1} - \left(y_{k+1} \times \frac{\Delta x}{\Delta y} + C \times \frac{\Delta x}{\Delta y} \right) \right]$$

$$= \Delta y \times \frac{\Delta x}{\Delta y} \left[(y_{k+1} - C) - x_k - x_{k+1} - \left(\frac{\Delta y}{\Delta y} \times \frac{\Delta x}{\Delta y} y_{k+1} + C \right) \right]$$

$$= \Delta x \Delta y \, \Delta x (y_{k+1} - C) - x_k - x_{k+1} - \Delta x \, y_{k+1} + C$$

Bresenham's Algorithm examples

① Digitize the line with end points (20,10), (30,18) using Bresenham's line algorithm.

soln,

Given:

$(x_1, y_1) = (20, 10)$

$(x_2, y_2) = (30, 18)$

we have,

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= \frac{18-10}{30-20}$$

$$= \frac{8}{10}$$

$$= 0.8 < 1$$

Then,

$\Delta x = x_2 - x_1 = 30 - 20 = 10$

$\Delta y = y_2 - y_1 = 18 - 10 = 8$

$2\Delta y = 2 \times 8 = 16$

$2\Delta x = 2 \times 10 = 20$

$2\Delta y - 2\Delta x = 16 - 20$

$= -4$

Initial decision parameter, since $m < 1$.

$p_0 = 2\Delta y - \Delta x$

$= 16 - 10$

$= 6,$

condition

if $m < 1$

if $p_k \geq 0$, $y_{k+1} = y_k + 1$, $x_{k+1} = x_k + 1$

$p_{k+1} = p_k + 2\Delta y - 2\Delta x$

if $p_k < 0$, $y_{k+1} = y_k$, $y_{k+1} = y_k + 1$

$p_{k+1} = p_k + 2\Delta y$

$p_0 = 2\Delta y - \Delta x$

if $m > 1$

if $p_k \geq 0$, $x_{k+1} = x_k + 1$, $y_{k+1} = y_k + 1$

$p_{k+1} = p_k + 2\Delta x - 2\Delta y$

$p_k < 0$, $x_{k+1} = x_k$, $y_{k+1} = y_k + 1$

$p_{k+1} = p_k + 2\Delta x$

$p_0 = 2\Delta x - \Delta y$

k (iteration)	p_k	(x_{k+1}, y_{k+1})
0	$p_0 = 6$ $\therefore p_k > 0$	$x_{k+1} = 20+1 = 21$, $y_{k+1} = 10+1 = 11$ (21,11)
1	$p_1 = 6+16-20 = 6-4 = 2 > 0$	$x_{k+1} = 21+1 = 22$, $y_{k+1} = 11+1 = 12$ (22,12)
2	$p_2 = 2-4 = -2 < 0$	$x_{k+1} = 22+1 = 23$, $y_{k+1} = y_k = 12$ (23,12)
3	$p_3 = -2+16 = 14 > 0$	$x_{k+1} = 23+1 = 24$, $y_{k+1} = 12+1 = 13$ (24,13)
4	$p_4 = 14-4 = 10 > 0$	$x_{k+1} = 24+1 = 25$, $y_{k+1} = 13+1 = 14$ (25,14)
5	$p_5 = 10-4 = 6 > 0$	$x_{k+1} = 25+1 = 26$, $y_{k+1} = 14+1 = 15$ (26,15)
6	$p_6 = 6-4 = 2 > 0$	$x_{k+1} = 27$, $y_{k+1} = 16$ (27,16)
7	$p_7 = 2-4 = -2 < 0$	$x_{k+1} = 27+1 = 28$, $y_{k+1} = y_k = 16$ (28,16)
8	$p_8 = p_k + 2\Delta y = -2+16 = 14 > 0$	$x_{k+1} = 29$, $y_{k+1} = 17$ (29,17)
9	$p_9 = 14-4 = 10 > 0$	$x_{k+1} = 30$, $y_{k+1} = 18$ (30,18)
~~10~~	~~$p_{10} = 10-4 = 6 > 0$~~	~~$x_{k+1} = 31$, $y_{k+1} = 19$ (31,19)~~

Digitize the line joining the points (2,3) and (8,12) using BLA (2076 Board)

Soln,

Given, $(x_1, y_1) = (2,3)$

$(x_2, y_2) = (8,12)$

we have,

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= \frac{12-3}{8-2} = \frac{9}{6} = 1.5 > 1$$

Then,

$\Delta x = x_2 - x_1 = 8-2 = 6$

$\Delta y = y_2 - y_1 = 12-3 = 9$

$2\Delta x = 6 \times 2 = 12$

$2\Delta y = 9 \times 2 = 18$

$2\Delta y - 2\Delta x = 18-12 = 6$

initial decision parameter, since $m > 1$

$p_0 = 2\Delta x - \Delta y$

$= 18 - 6 = 12 - 9 = 3$

k (iteration)	p_k	(x_{k+1}, y_{k+1})
p0	$p_0 = 3 \quad \because p_k > 0$	$x_{k+1} = 2+1 = 3,\ y_{k+1} = 3+1 = 4 \quad (3,4)$
p1	$p_1 = 3 + 12 - 18 = -3 < 0$	$x_{k+1} = x_k = 3,\ y_{k+1} = y_{k+1} = 4+1 = 5 \quad (3,5)$
p2	$p_2 = -3 + 12 = 9 > 0$	$x_{k+1} = 3+1 = 4,\ y_{k+1} = 5+1 = 6 \quad (4,6)$
p3	$p_3 = 9 - 6 = 3 > 0$	$x_{k+1} = 5,\ y_{k+1} = 7 \quad (5,7)$
p4	$p_4 = 3 - 6 = -3 < 0$	$x_{k+1} = 5,\ y_{k+1} = 8 \quad (5,8)$
p5	$p_5 = -3 + 12 = 9 > 0$	$x_{k+1} = 6,\ y_{k+1} = 8+1 = 9 \quad (6,9)$
p6	$p_6 = 9 - 6 = 3 > 0$	$x_{k+1} = 7,\ y_{k+1} = 10 \quad (7,10)$
p7	$p_7 = 3 - 6 = -3 < 0$	$x_{k+1} = 8,\ y_{k+1} = 11 \quad (7,11)$
p8	$p_8 = -3 + 12 = 9 > 0$	$x_{k+1} = 8,\ y_{k+1} = 12 \quad (8,12)$

Q Digitize the line with end points (6, 12) and (10, 5) using BLA.

Soln,

Given, $(x_1, y_1) = (6, 12)$

$(x_2, y_2) = (10, 5)$

we have,

$$m = \left| \frac{y_2 - y_1}{x_2 - x_1} \right|$$

$$= \left| \frac{5 - 12}{10 - 6} \right| = \left| \frac{-7}{4} \right| = 1.75 > 1$$

(Here, $y_1 > y_2$, so y is decrement is successive state)

$\Delta y = y_2 - y_1 = 5 - 12 = -7$

$\Delta x = x_2 - x_1 = 10 - 6 = 4$

$2\Delta x = 2 \times 4 = 8$

$2\Delta y = 2 \times (-7) = -14$

initial decision parameter since $m > 1$

$p_0 = 2\Delta y - \Delta x$

$= -14 - 4$

$= -18$

$p_0 = 2\Delta x - \Delta y$

$= 8 - 7$

$= 1$

k (iteration)	p_k	x_{k+1}, y_{k+1}
0	$p_0 = 10 > 0$	$x_{k+1} = x_k - 6,\ y_k + 1 =$
1	$p_1 = 10 + 8 - 14 = 4 > 0$	$x_{k+1} = x_k - 6 + 1 = 7,\ y_{k-1} = 12 - 1 = 11$ (7,11)
2	$p_2 = 4 - 6 = -2 < 0$	$x_{k+1} = 8,\ y_k - 1 = 10$ (8, 10)
3	$p_3 = -2 + 8 = 6 > 0$	$x_{k+1} = 9,\ y_{k-1} = 10$ (9, 10)
4	$p_4 = 6 - 6 = 0 \geq 0$	$x_{k+1} = 11,\ y_{k-1} = 8$ (11, 8)

k (iteration)	p_k	x_{k+1}, y_{k+1}
0	$p_0 = 1 > 0$	$x_{k+1} = x_k - 6,\ y_k + 1 = 11$ (7,11)
1	$p_1 = 1 + 8 - 14 = -5 < 0$	$x_{k+1} = 7,\ y_{k-1} = 10$ (7,10)
2	$p_0 = -5 + 8 = 3 > 0$	$x_{k+1} = 8,\ y_{k-1} = 9$ (8, 9)
3	$p_0 = 3 - 6 = -3 < 0$	$x_{k+1} = 8,\ y_{k-1} = 8$ (8, 8)
4	$p_0 = -3 + 8 = 5 > 0$	$x_{k+1} = 9,\ y_{k-1} = 7$ (9, 7)
5	$p_0 = 5 - 6 = -1 < 0$	$x_{k+1} = 9,\ y_{k-1} = 6$ (9, 6)
6	$p_0 = -1 + 8 = 7 > 0$	$x_{k+1} = 10,\ y_{k-1} = 5$ (10, 5)

Algorithm of mid-point circle Drawing?

1. Input radius 'r' and circle center (x_c, y_c) & 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 $p_k < 0$, the next point along the circle centered on $(0,0)$ is (x_{k+1}, y_k) and

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

else, the next point is $(x_k + 1, y_k - 1)$ &

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

4. Determine symmetry points in the other 7 octants.
5. Move each calculated pixel position (x, y) onto the circular path centered on (x_c, y_c) & plot the co-ordinates values.

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

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

Advantages of Bresenham's line Algorithm (BLA) over DDA.

- In DDA Algorithm each successive point is computed in floating point. so, it required more time & memory space. while in BLA, each successive point is calculated in integer value or whole number. so it require less time & less memory.

- In DDA, since the calculated point value is floating point number, it should be rounded at the end of calculation but in BLA, it doesn't need to round. So, there is no accumulation of rounding error.

- Due to rounding error, the line drown by DDA algorithm is not accurate, while in BLA line is accurate.

- DDA can't be used in other application except line drowing, but BLA can be implemented in other application such as circle, ellipse & other curves.

BLA Algorithm for $|m| < 1$.

1. Input the two line end points & store the left end point in (x_0, y_0)
2. load (x_0, y_0) into the frome buffer i.e. plot the first point.
3. calculate constants Δx, Δy, $2\Delta y$ & $2\Delta y - 2\Delta x$ and obtain the starting value for the decision parameter as

$$p_0 = 2\Delta y - \Delta x$$

4. At each x_k, along the line, starting at $k = 0$, perform the following tests.

If $p_k < 0$, then next point to plot is (x_{k+1}, y_k) &

$$p_{k+1} = p_k + 2\Delta y$$

otherwise, the next point to plot is (x_{k+1}, y_{k+1}) &

$$p_{k+1} = p_k + 2\Delta y - 2\Delta x$$

5. Repeat step 4 for Δx times.

BLA Algorithm for $|m| \geq 1$

1. Input the two line ~~segments~~ end points & store the left end point in (x_0, y_0)
2. load (x_0, y_0) into the frame buffer i.e. plot the first point.
3. calculate constants Δx, Δy, $2\Delta y$ & $2\Delta y - 2\Delta x$ and obtain the starting value for the decision parameter as

$$p_0 = 2\Delta x - \Delta y$$

4. At each y_k along the line, starting at $k=0$, perform the following tests.

If $p_k < 0$, then next point to plot is $(x_k, y_k + 1)$

& $p_{k+1} = p_k + 2\Delta x$

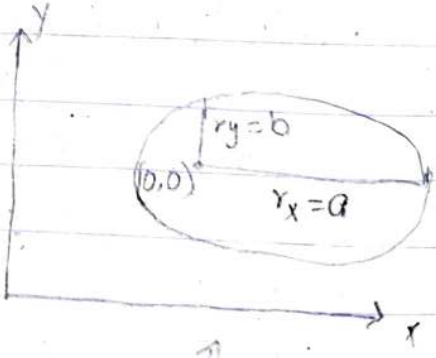
otherwise, the next point to plot is $(x_k + 1, y_k + 1)$

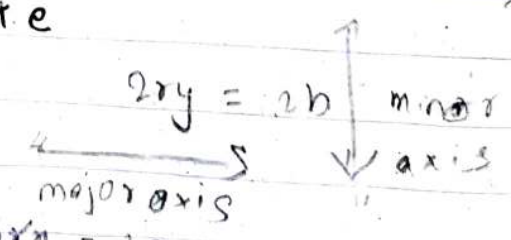
$$p_{k+1} = p_k - 2\Delta y + 2\Delta x$$

5. Repeat step 4 for ~~Δx~~ Δy times.

Ellipse

- Ellipse is an elongated circle. Therefore elliptical curves can be generated by modifying circle drawing procedures to take into account the different dimensions of an ellipse along the major & minor

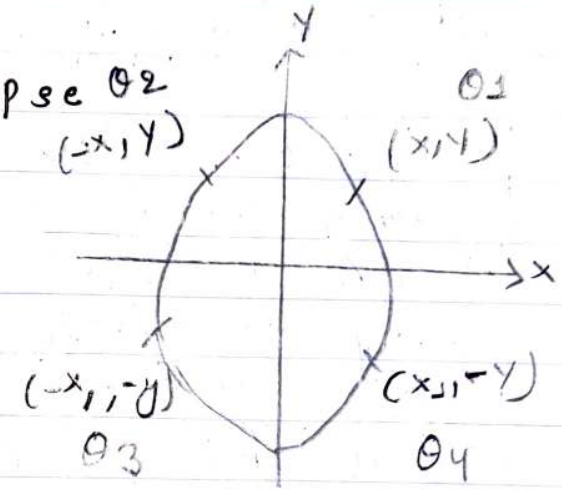




axes.

- The general equation of ellipse with semi-major axis $r_y = b$ centered at (x_c, y_c) is

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

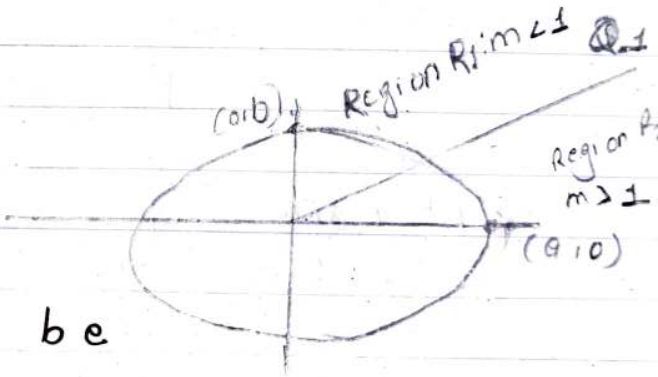


since,

$$(x_c, y_c) = (0, 0)$$

$$\frac{x^2}{r_x^2} + \frac{y^2}{r_y^2} = 1$$

$$\therefore \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \quad \text{---(ii)}$$



In polar form, the eqn can be represented as

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

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

Symmetry of Ellipse

- calculation of a point (x, y) in one quadrant yields the yields the ellipse point in other 3-quadrant.

For Region R_1, we take.

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

$$\therefore m < 1$$

For region R_2, we take

$$y_{k+1} = y_k - 1$$

$$\therefore m \geq 1$$

mid-point Ellipse Algorithm :-

1. Input ellipse center (x_c, y_c), semi major-axis length $r_x (a)$ & minor-axis length $r_y (b)$ & obtain the first point on ellipse centered at origin as $(x_0, y_0) = (0, b) = (0, r_y)$

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

$$p1_0 = b^2 - a^2 b + 1/4\, a^2$$

3. At each step x_k position in region 1, starting at $k = 0$, perform the following test.

If $p1_k < 0$, then the next point on ellipse centered at $(0,0)$ is

$$(x_k + 1, y_k) \text{ & } p1_{k+1} = p1_k + 2b^2 x_k + 1 + b^2$$

otherwise,

$$x_k + 1, y_k - 1$$

$$\therefore p1_{k+1} = p1_k + 2b^2 x_k + 1 - 2a^2 y_k - 1 + b^2$$

with,

$$2b^2 x_{k+1} = 2b^2 x_k + 2b^2$$

$$2a^2 y_k - 1 = 2a^2 y_k - 2a^2$$

And continue until,

$$2b^2 x \geq 2a^2 y$$

4. Calculate the initial value of the decision parameter

in region 2 using line last point (x_0, y_0) calculated in region 1

$$p2_0 = b^2 (x_0 + 1/2)^2 + a^2 (y_0 - 1)^2 - a^2 b^2$$

5. At each step y_k in region 2, starting at $k = 0$, perform the following test.

If $p^2_k > 0$, the next point along the ellipse centered on $(0,0)$ is $(x_k, y_k - 1)$ &

$$p^2_{k+1} = p2_k - 2a^2 y_{k+1} + a^2$$

Otherwise,

the next point is $(x_k + 1, y_k - 1)$ &

$$p^2_{k+1} = p2_k - 2b^2 x_{k+1} - 2a^2 y_{k+1} + a^2$$

6. Determine symmetry points in the other 3-quadrant.

7. Move each calculated pixel point position (x, y) onto the elliptical path centered on (x_c, y_c) & plot the co-ordinate values.

$$x = x + x_c$$

$$y = y + y_c$$

8. Repeat the step for region 1 until $2b^2 x \geq 2a^2 y$ & Region 2 until $(a, 0)$

2 Dimension Transformation (x, y)

Changing co-ordinates description of an object is called Transformation. changes in orientation, size & shape are accomplised with geometric Transformation. Types :-

i) Rigid body transformation: Transformation without changes or deformation in shape

ii. non-Rigid body Transformation: Transformation with changes in shape.

when a transformation takes place on 2-D plane, it is called 2D-Transformation.

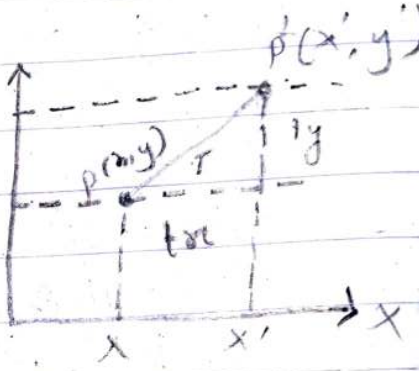
2 Dimension Transformation (x, y)

Types:

i. Translation - position
ii. Rotation - Angle
iii. scaling - size

2D Translation

~~Changing co-ordinates des~~



Changing the co-ordinates position along a straight line is called Translation. Repositioning of object along a straight-line path from one co-ordinate location to another is called translation.

- Translation is performed on a point by adding offset to its co-ordinate so as to generate a new co-ordinate position.

- let P(x, y) is translated to position P'(x', y') by translation distance tx units parallel to x-axis & ty units distance to y-axis. Then,

$$x' = x + tx$$
$$y' = y + ty$$

In matrix form,

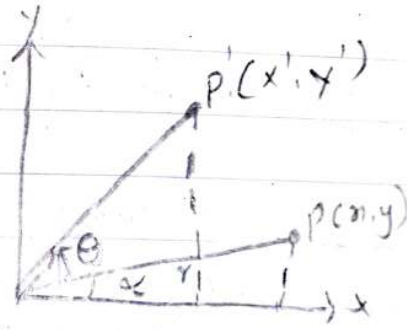
$P' = P \cdot T$ where T is transformation matrix

i.e. $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} tx \\ ty \end{bmatrix}$

(iv) 2D - rotation

- Changing the co-ordinate position along a circular path is called rotation.
- 2D rotation is applied to re-position the object along a circular path in xy-plane. Rotation is generated by specifying rotation angle (θ) and rotation point (pivot point).
- The positive θ rotates object in anti-clockwise direction.
- The negative θ rotates object in clock-wise direction.

Types of rotation
- About origin
- About any pivot point



i. About origin

- If P (x,y) is rotated in a new point P'(x',y') in anti-clockwise direction by an angle θ. then (x',y') can be calculated in following ways:

- let the angle made by line op with x-axis is 'α' & the radius of circular path is 'r'. Then,

$$x = r\cos\alpha$$
$$y = r\sin\alpha$$

Also,

$$x' = r\cos(\theta + \alpha)$$
$$= r\cos\theta\cos\alpha - r\sin\theta\sin\alpha$$
$$x' = x\cos\theta - y\sin\theta$$

Similarly,

$y' = r \sin(\theta + \alpha)$

$= r \sin\theta \cos\alpha + r \cos\theta \sin\alpha$

$\therefore y' = x \sin\theta + y \cos\theta$

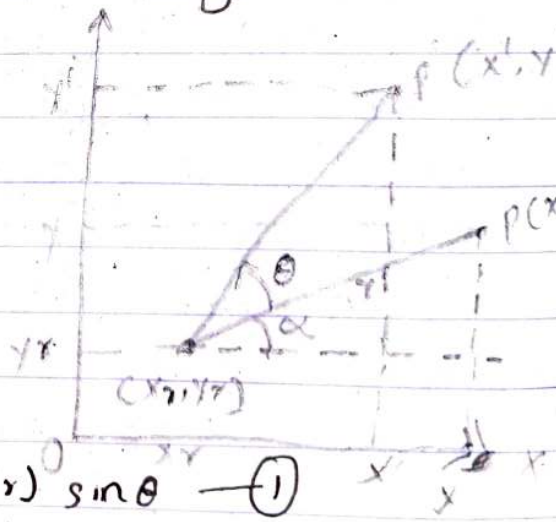
In matrix form,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$\therefore P' = R \cdot P$

ii About Any pivot point (x_r, y_r)

- If (x, y) is rotated to new position $P'(x', y')$ by an angle θ & taking any point (x_r, y_r). Then (x', y') can be calculated as following.



$$\cos(\theta + r) = \left(\frac{x' - x_r}{r}\right)$$

or, $r\cos(\theta + r) = x' - x_r$

or, $x' - x_r = r\cos\theta\cos\alpha - r\sin\theta\sin\alpha$

since, $r\cos\alpha = x - x_r$,

$r\sin\alpha = y - y_r$

$\therefore x' = x_r + (x - x_r)\cos\theta - (y - y_r)\sin\theta$ — (1)

similarly,

$$\sin(\theta + \alpha) = \left(\frac{y' - y_r}{r}\right)$$

or, $r\sin(\theta + \alpha) = y' - y_r$

or, $y' - y_r = r\sin\theta\cos\alpha + r\cos\alpha\sin\theta$ ~~$r\cos\alpha\sin\theta$~~ $r\sin\alpha\cos\theta$

a

since, $r\cos\alpha = x - x_r$,

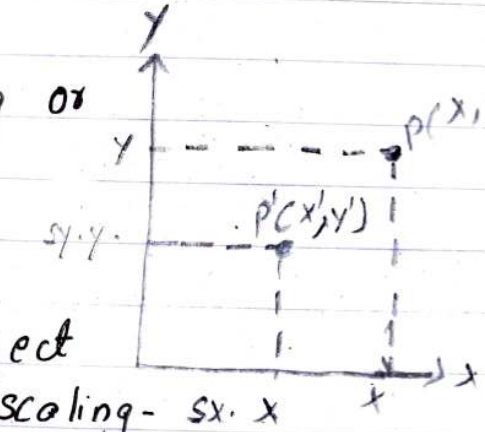
$r\sin\alpha = y - y_r$

$\therefore y' = y_r + (x - x_r)\sin\theta + (y - y_r)\cos\theta$ — (2)

These equation (1) & (2) are the equation for rotation of a point (x, y) with angle θ taking pivot point (x_r, y_r).

(3) 2D - Scaling

- Scaling is the process of expanding or compressing the dimension of an object.



- The co-ordinate position of an object by multiplying a constant factor (scaling factor) is called scaling.
- It alters the size of an object.
- A simple two-dimensional scaling operation is performed by multiplying object position (x, y) with scaling factor s_x & s_y along x & y direction to produce (x', y')

i.e. $x' = s_x \cdot x$

$y' = s_y \cdot y$

In matrix form,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- If the scaling factor are in betⁿ 0 & 1, point is closer to origin & the size of the object decreases.
- If the scaling factor are greater than 1, point is

away from origin & the size increases.

- If (Sx, Sy) = 1, for both direction doesn't change the size of object & this scaling is called uniform scaling.
- If Sx & Sy are different, then this scalling is called differential scaling which is widely used in graphical package to change the shape of the object.

graphic.h ✓
Stdio.h ✓
console.h
conio.h ✓

graphics.h - Declares prototypes for the graphics function.

Stdio.h - Defines types and macros needed for the standard I/O package defined in Kernighan and Ritchie and extended under UNIX system V. Defines the standard I/O predefined streams stdin, stdout, stdprn, and stderr, and declares stream-level I/O routines.

~~console.h~~

conio.h

Declares various functions used in calling the DOS console I/O routines.

functions of Graphics

arc	bar
circle	inligraph
ellipse	
bar	
clear	

~~gd:~~

① bar

Declaration: void bar(int left, int top, int right, int bottom);

gd = graphics doiver

gm = graphic mode

Unit - 3

Three dimensional Graphics (3D)

*

Introduction :

Difference between parallel & prospective projection

edge table

E1 = V1 V6 E2 = V2 V8 E3 = V6 V4 E4 = V4 V5

E5 = V1 V2 E6 = V2 V3 E7 = V7 V3 E8 = V5 V6

E9 = V1 V7 E10 = V5 V7

S1 = E1, E11, E12, E3

S2 = E12, E9, E8, E10

S3 = E5, E6, E8, E7

S4 = E1, E2, E5, E4

Introduction of lighting model

Illumination model / shading Model

- Once visible surface has been identified by hidden surface algorithm, a shading model / lighting model / illumination model is used to complete the intensities & color to display for the surface.

- Illumination model is the model for calculating light intensity at a single surface point. An Illumination model also called as lighting model & sometime referred to as a shading model is used to calculate the intensity of the light that we should see at a given point on the surface of an object.

- It is used to calculate the color of an illuminated position on the surface of an object.

- A surface rendering algorithm uses the intensity calculations from an illumination model.

- An object is illuminated from the ambient light (background) & from interrelated light sources.

object illuminate on the basis of following properties:

- Intensity of Ambient light.
- Types of object surface
- Surface color and location.

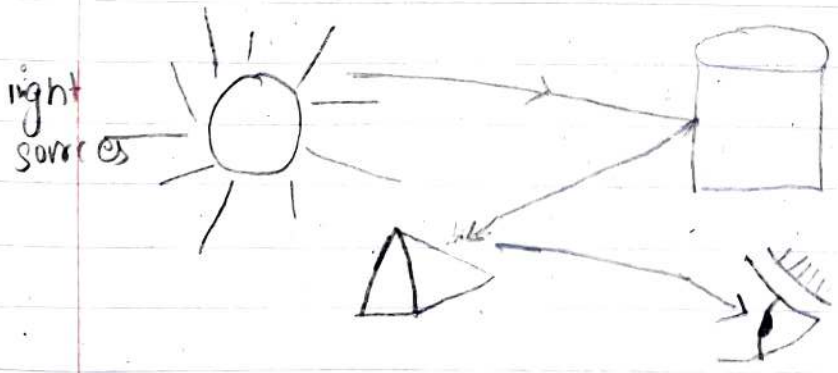
Components of Illumination Model

① light source:

object that radiates energy are called light sources such as sun, lamp, bulb, fluorescent lube etc. sometime light sources are reffered as light, emitting object or light reflectors.

- Generally light source is used to mean an object that that is emitting radiant energy for eg: sun

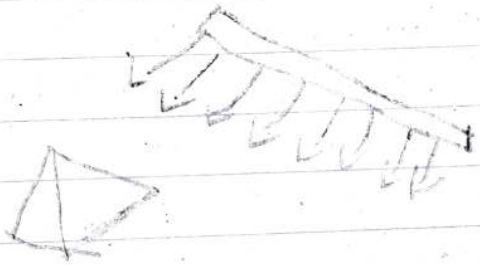
- Total reflected light = contribution from light sources + contribution from reflecting surfaces



(i) Distributed source

It is as if the light source was infinitely far way from the surface that is illuminating.

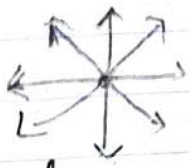
- Area of source is not small compared to the surfaces in the scene. for eg: fluorescent lamp/tube.
- All of the rays from a directional or distributed light source have the same direction & no point of origin.



(ii) point source

It is the simplest light emitter eg. light bulb.

- It transmit beam of light in all directions from a fixed point location whose dimensions are small compared to the size of the object in a scene.



- It consist point of origin.

(iii) parallel source

A ~~parallel~~ particular region which is far from the substrate can be called (the sun).

Reflection of light

when light is incident on opaque surface part of it is reflected & part of it is absorbed

$I = A + R \longrightarrow$ Reflected light.

incident light Absorbe light.

- The amount of incident light reflected by a surface depends upon the type of material.

& dull surfaces absorbs more light.

- shining material reflects more light. for transparent surface, some of light will be reflected & some will be transmitted through the material.

i. Diffuse Reflection.

surfaces that are rough or grainy tend to scatter the reflected light in all direction. This scattered light is called diffuse reflection.

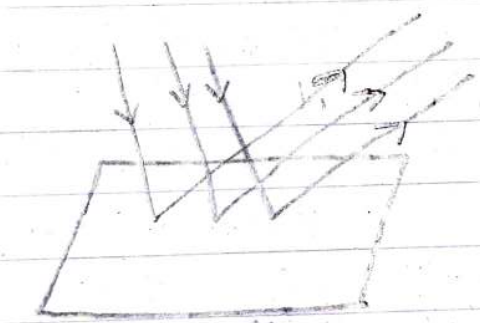
- The surface appears equally bright from all viewing direction when rough matte surface produces primarily diffuse reflection.

(ii) Specular Reflection

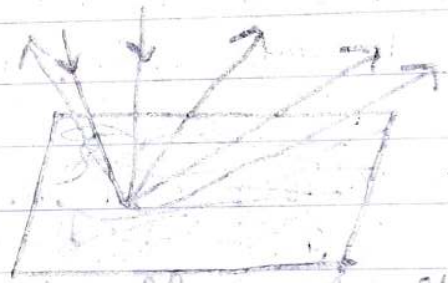
In addition to diffuse reflection, light sources create highlights or bright spots called specular reflection.

- This highlighting effect is more pronounced on shiny surfaces than on dull surface.

- As light passes, portion of it is transmitted on the substrate & most of it is consumed.



specular reflection



diffuse reflection

Illumination model

Illumination model are used to calculate the light intensities that we should see at a given point on the surface on an object. lighting calculations are ~~used~~ based on the ~~opt~~ optical properties of surfaces, the background lighting conditions & the light sources specification

- All light sources are considered to be point sources, specified with a coordinate position &

an intensity value (color).
some illumination model are as follows:

1. Ambient light Illumination model
2. Diffuse Reflection Illumination model
3. specular Reflection Illumination model

① Ambient light Illumination model (light coming from the nearby object after reflection)

This is the simplest way to model the combination of light reflections from various surfaces to produce a uniform illumination called ambient light or background light.

- A surface that is not exposed directly to light source still will be visible if nearby objects are illuminated.
- Equal amount of light from all directions. i.e The amount of ambient light incident on each object is a constant for all surfaces & over all directions
- Ambient light means the light that is already present in a scene, before any additional lighting is added,
- It usually refers to natural light., either outdoors or coming through windows etc. It can also mean artificial light such as normal room lights.
- The intensity of reflected light for each surface depends on the optical properties of the surface (shinny, rough), that is how much of the incident light is to be reflected & how much absorbed.

- we can set the level for the ambient light in a scene with parameter I_{amb} & each surface is then illuminated with this constant value i.e.

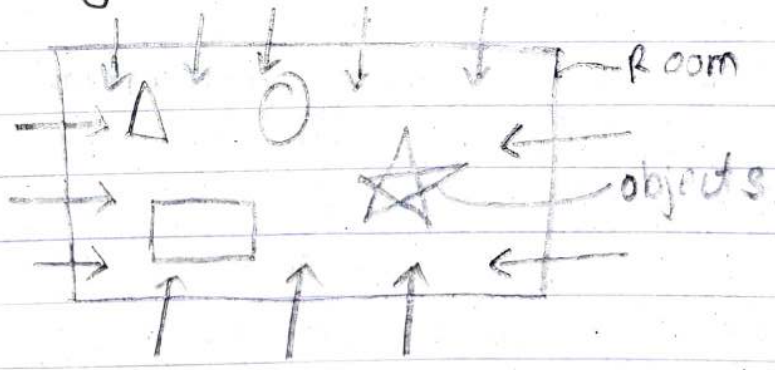
$$I_{amb} = K_a I_a$$

where,

I_a = ambient light intensity which describe the level of ambient light.

K_a = percentage of the light reflected by the object's surface which ranges from 0 to 1.

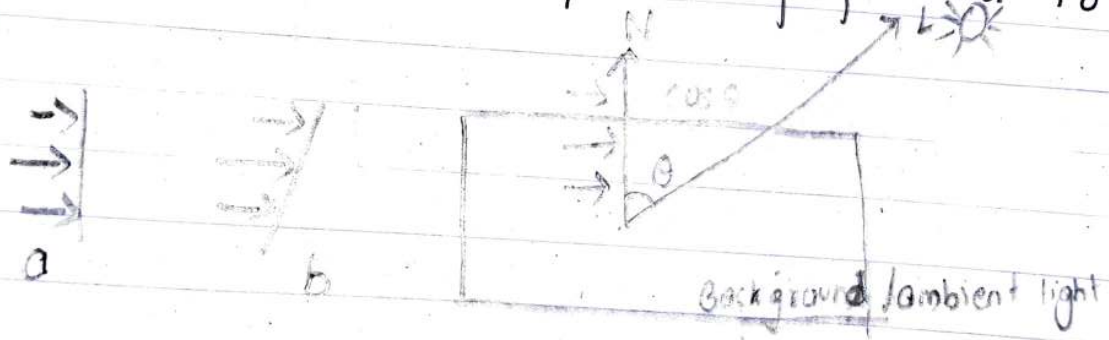
- Independent of surface orientation & viewer location. But the different surface way reflect different amount of ambient light.



(2) Diffuse Reflection Illumination model :

The light that is reflected in all directions is called diffuse reflection. The reflected light is independent of viewing position because it is equally bright from all the directions. But depends upon the surface orientation.

- The objects brightness varies from one part to another depending on the direction of and distance to the light sources.
- color of an object is determined by the color of ~~diffuse~~ diffuse reflection of the incident light.
- If any object surface is red then there is a diffuse reflection for red component of light & all other components are absorbed by the surface.
- The brightness at each point is proportional to $\cos\theta$



If we denote the angle of incidence betn the incoming light direction & the surface normal as 'θ', then the projected area of a surface path perticular to the light direction is proportional to $\cos\theta$.

- If N is the unit normal vector to a surface & L is the unit direction vector to the point light source from a position on the surface. Then,

$$\cos\theta = N \cdot L$$

and the diffuse reflection equation for single point source Illumination is given by

$I_{diff} = k_d I_p \cos\theta = k_d I_p (N \cdot L)$

where,

I_p = point of light intensity

k_d = Diffuse reflectivity of surface, value of $k_d = 0$ to 1

N = normal surface

L = direction of light.

LAB-2

#1 WAP to draw a bar & circle with co-ordinate.

```
#include <stdio.h>
#include <conio.h>
#  inclide <graphics.h>
  void main()
{
int gd = DETECT, gm;
initgraph (&gd, &gm, "C:\\turbo c3\\bgi");
   bar (100, 100, 200, 200);
    circle (300, 150, 50)
      getch ();
    close graph ();
        }
```

LAB 3

Display text on screen:

outtext xy (int x, int y, char *string)

```
int main ()
{ int gd = DETECT , gm;
int i, x = 100,  y = 50;
initgraph (&gd, &gm, "C:\\turboc3\\bgi");
for(i = 1, i<16; i++);
 {
set color (i);
outtext xy ( x, y, " welcome to Basv ");
   y=y+30;
   }
getch ();
closegraph ();
return 0;
}
```

Shading model :

- In computer Graphics shading is referred to the process of altering the color of object, surface, polygon in 3D - scene based on the surface angle to light, it's distance from lights & material properties etc.

- Shading is referred to as the implementation of the illumination model at the pixel points or polygon surfaces of the graphics object.

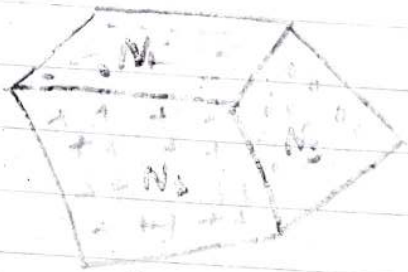
- Shading model is used to compute the intensities & colors to display the surface.

(1) Constant Intensity Shading / Flat Shading

- A fast & straight forward method of rendering an object with polygon surfaces is constant intensity shading, also called flat shading.

- In this method, a single intensity is calculated for each polygon and the intensity is applied to all points of the surface of the polygon. Hence, all points over the surface of the polygon are then displayed with the same intensity value.

- This technique display all points in polygon with single shading.
- constant shading can be useful for quickly displaying the general apperance of the curved surface as shown in figure.



In general, flat shading of polygon facets provides an accurate rendering for an object if all the following assumption is valid.

① The object is a polyhendron & is not an approximation of an object with curved surfaces.

② All light sources is at infinily (N·L is constant on polygon)

③ The viewing position is sufficiently far from the surface so that V·R is constant over the surface.

Advantages
- useful for quick display of curved surface.

Dis-advantages
- Discountinuity of colours can be observed in different faces of colors of objects.

Gouraud shading

- This intensity-interpolation schema, developed by Henri Gouraud & usually referred as Gouraud shading, renders a polygon surface by linear interpolating intensity value across the surface.

- Intensity values for each polygon are matched w/ co-ordinate with the value of adjacent polygon along the common edge thus, eliminating the intensity discontinuities that can occurs in flat shading.

- each polygon surface is rendered with Gouraud shading by performing the following calculations.

① Determining the average ~~wit~~ unit normal vector at ~~each~~ each polygon vertices.

② Apply an illumination model to each vertex to determine the vertex intensity.

③ linear interpolate the vertex intensities over the surface of the polygon.

"C:\TURBOC3\include ... h"

DDA line Drawing Algorithm

```
#include <conio.h>
#include <stdio.h>
#include <graphics.h>
void main()
{
int gd, gm;
int x1 = 0, x2 = 100, y1 = 0, y2 = 100, step, dx, dy, xinc,
        yinc, i;
detectgraph (&gd, &gm);
initgraph (&gd, &gm, "C:\\Turbo C++\\bgi");
    dx = x2 - x1;
    dy = y2 - y1;
  if (dx >= dy)
        step = dx;
else
  step = dy;
xinc = (float) dx / step;
yinc = (float) dy / step;
for (i = 1; i < step; i++)
{
 putpixel (x1, y1, RED);
  x1 = x1 + xinc;
  y1 = y1 + yinc;
}
getch ();
closegraph ();
}
```

Unit - 5

Fundamental of Animation Techniques

* *

Animation :-

It is defined as the act of making something come alive.

It refers to the movement on the screen of the display device created by displaying a sequence of still images.

It is the process in which the illusion of movement is achieved by creating & displaying a sequence of images with elements that appear to have motion.

- In animation, a series of images rapidly changed to create an illusion of movement.
- Animation is the technique of designing, drawing, making layouts & preparation of photographic series which are integrated into the ~~mult~~ multimedia & gaming products.
- some general application of computer animation are animated cartoons, movies, logos, advertising, scientific & engineering studies, conventional animation (2D) & raster Animation (3D).
- A person who creates animation is called Animator

(person)

He/she use various computer technologies to capture the pictures and then to animate these in the desired sequence.

Application

① Education & Training:

Animation is used in school, college & training centres for education purpose. Flight simulators for aircraft are also animation based.

② Entertainment:

Animation methods are now commonly used in making motion pictures, music videos, & television shows, etc.

③ Computer Aided Design (CAD):

One of the best applications is CAD. One of the earlier applications of CAD was automobile designing. But now almost all types of designing are done by using CAD application, and without animation all works can't be possible.

④ Advertising:

This is one of the significant applications & is the most important advantage of an animated advertisement is that it takes very less space & capture people attention.

(5) Artistic purpose, data displaying in a scientific visualization, story telling & Instructional purpose etc.

Types of Animation

① Cel Animation

- A traditional forms of animation used in the production of cartoons and animated movies where each frame of the scene is drawn by hand.
- A full length features film produced using cel animation would often require a million or more drawing to complete.

② Computer Animation

→ subset of both computer graphics & animation Technology.
→ It is the creation of moving images (animation) using computer technology.

③ Kinematics

- It is the study of the movement & the motion of structures that have joints, such as walking persons.
- such types of animation are usually used in the areas like mechanics etc.

(4) Morphing

- It is popular effect in which one image transforms into another.
- Change the shape. size, color etc.
- The morphed images were build at a rate of 8 frames per seconds, with each transition taking a total of 4 seconds.

for eg: Face look at young Age, old age & teenage or present time.

Animation file format & software

Animation file format

- Director *.dir
- Animation pro *.fli, *.flc
- 3D studio max *.max
- superc ard & Director *.pics
- Compuserve *.gif
- flash *.flo, *.swf

software used

- 3D studio max, flash, Animation pro.

V. Imp

Designing an Animation sequence.

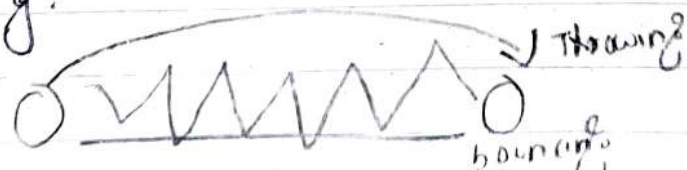
1. story board layout
2. object Description / definition
3. key-frame specification
4. Generation of in between frames
5. line testing / Editing
6. Recording

} main steps

① story board layout (Requirement)

- The story board is an outline of the action. This defines the motion sequence as a set of basic events that are to take place.

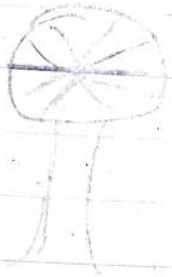
- A story board is a physical or digital visual overview of a project.
- Depending on the types of animation to be produced, the storyboard could consist of a set of rough sketches or it could be a list of a basic ideas for the motion.
- It consist of a set of key sequences- for eg:



Throwing the ball or bouncing the ball to reach another place.

② object Description / definition

- Each object participating in the action is given object definition.
- The object can be described in terms of basic shapes such as lines, curves, polygon or splines etc.
- The object are developed using curve, surface or solid representation. A great degree of realism can be added to these object by shading. The shading attributes ~~could~~ be color, texture, reflectance etc.



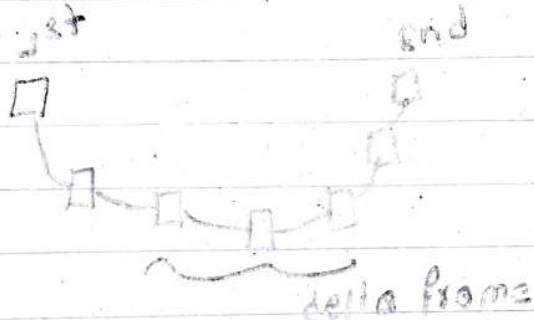
In eg: mushroom shaded.

③ key-frame specification

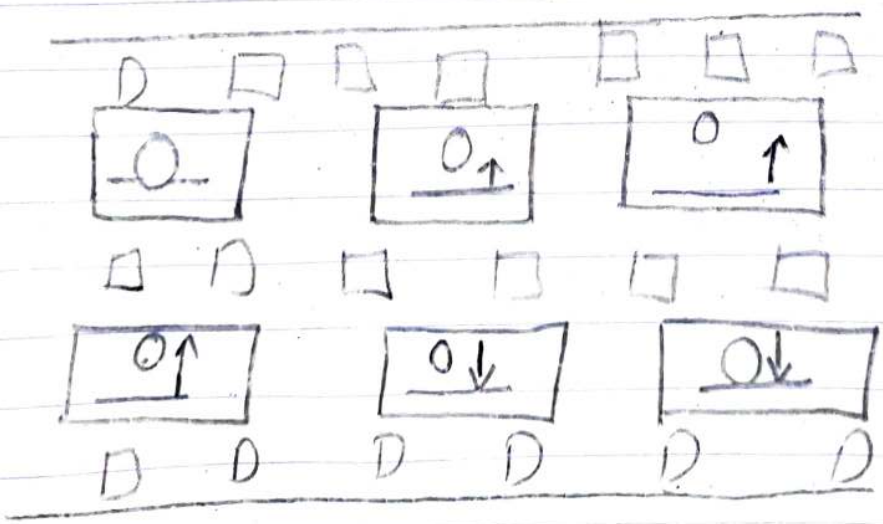
- Frame is one from any photographic picture in moving form or state.
- A keyframe in animation & film making is a drawing that defines the starting & ending point of any smooth transition.
- A sequence of keyframe which movement the spectator will see, but the position of the keyframes on the film defines the timing of the movement. 2 or 3 can be present for a spam of a second.
- Also called as In betn frames.

- It contains sequence of keyframes which defines movement of viewer eyes.
- It is detailed drawing of a scene at a given animation sequence within each keyframe, each object is positioned according to the time for that frame. More key frames are specified for intricate (very complicated or detailed) motion.

for eg: flower blossome.



- 2 or 3 delta frame can be present for a spam of second betn key frames.



(4) Generation of in betn frames

It is the process of generating Intermediate frames betn 2 images to give appreance that the first, image evolves smoothly into the second image.

- In betn are the drawing betn the keyframe which helps to create the illusion of motion
- Generation of in betn frames include the interporation of many frames betn 2 keyframes to produce smooth motion.
- film requires 24 frames per second. If animation is for n seconds, the in beth frames must be 24 n of the same to complete this movement.
- Graphics terminals at the rate of 30-60 per second

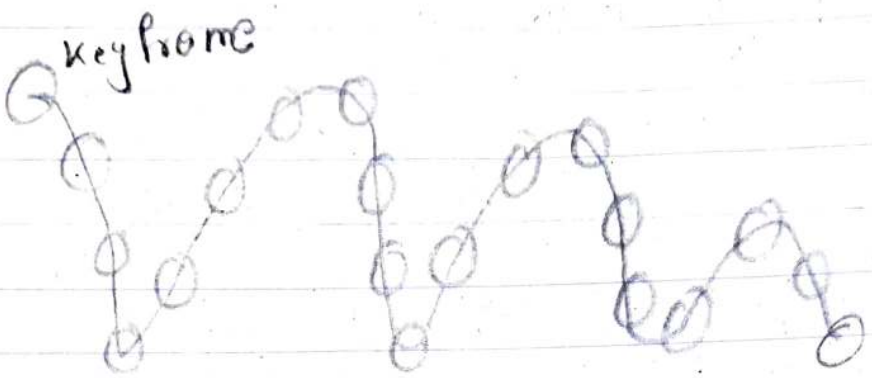
A part from above sequence, other task requirement are motion verification, editing, line testing & recording.

- line testing is used to check the quality of movement produced.

-> Recording is carried out frame by frame on to video recorders or films when all the frames & images are found satisfactory.

P.T.O

for eg:



In both frames generated by computer which creates illusion of motion.

computer Animation language

1. key frame system
2. Animation description
3. parameterized system
4. scripting system

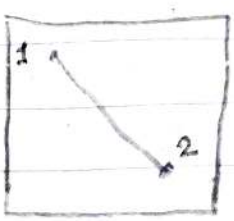
① key frame system

key frame is a frame where we define changes in animation. Every frame is keyframe when we create frame by frame animation.

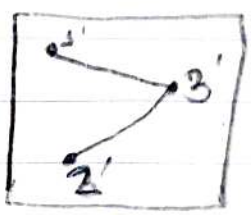
when someone create a 3D animation on a computer, they usually don't specify the exact position of any given object on every single frame. They create keyframes.

In it each object scene is defined as a set of rigid bodies connected at the joints & with a limited no. of degrees of freedom.

- whenever key frames are specified, it generates in betn frame specified in the form of kinetic description. i.e. we can use set of splines. we can use physical description also where we can use how much force we have used.
- we need to know about morphing i.e. changing the object from one shape to another.
- Motion path can be given with a kinematic description as a set of spline curves, or the motion can be physically based by specifying the forces acting on the object to be animated.
- For complex scene, we can separate the frames into individual components or object. Given the animation path, we can interpolate the position of individual object betn any two times.
- with complex object transformation the shapes of object may change over time. for eg: clothes, facial features, magnified detail, evolving shapes, exploding or disintegrating objects, transforming one object to another.



Keyframe k



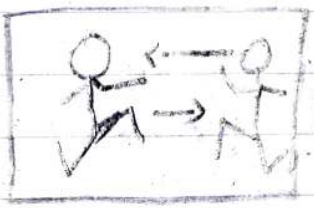
Keyframe k+1

- An edge vertexposition 1 & 2 keyframe k evolves into two connected edge in keyframe k+1 as 1', 2' & 3'.

parameterized system:

- It allows object motion characteristics to be specified as part of the object definitions.

- The adjustable parameters control such object characteristics as degree of freedom, motion limitations & allowable shape changes.
for eg: morphing in advertising



Two object when move & change, the parameters should not effect each other this is the degree of freedom. whereas object should move within the frame & can't move out of frame.

V. Imp

* Morphing

- It is animation that changes one image shape slowly to another image.
- It is often used to change one form to face to another.
- To seemly change from one image to another image a series of point is defined in both the before & after object.
- Then the pixel in the first object are slowly change

in the pixel in the second object by creating in betn frames with only changes from the previous change.

- Transformation of object shapes from one to another is called morphing, which is short form of metamorphosis. This method can be applied to any motion or transition involving a change in shape.

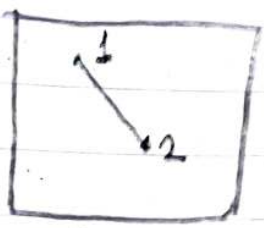
for eg: Human face is converted into animal face.
Face of young person ~~one~~ converted into aged person etc.

(#) The process of morphing involves the 3-steps.

(1) In the first step, one initial image & other final image are added to morphing application as shown in figure 1st & 4th object consider as keyframes.

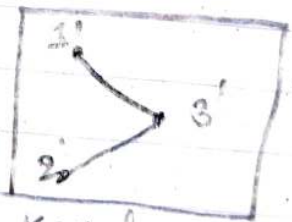
(2) The second step involves the selection of key points on both the images for a smooth transition betn two object / images ~~or~~ as shown in 2nd object.

(3) In the 3rd step, the keypoint of the first image transforms to a corresponding key point of the second image as shown in 3rd object in the figure.



Key frame k

In fig, an edge with vertex position 1 & 2 in key frame k evolves into two connected edge in key frame k+1.



key frame k+1

In fig, linear interpolation for transformation a line segment in to two connected line segments frames.

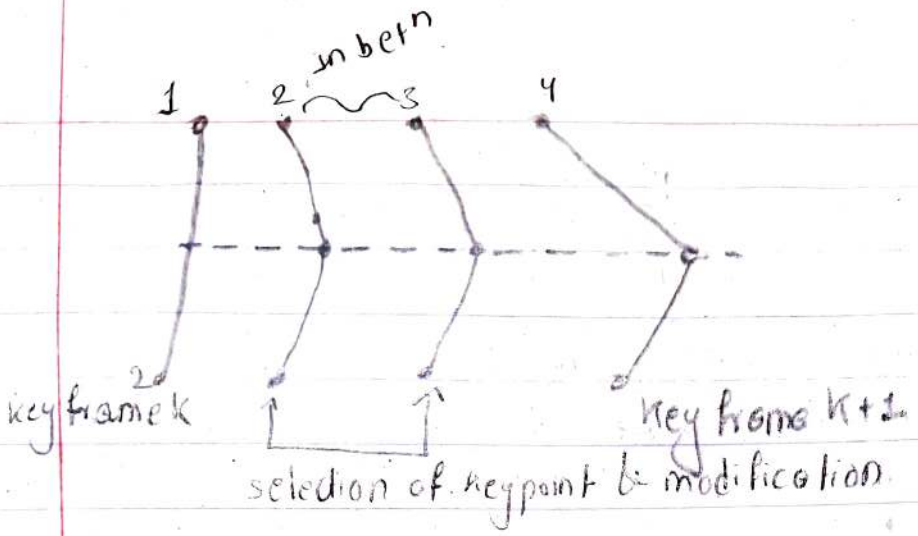


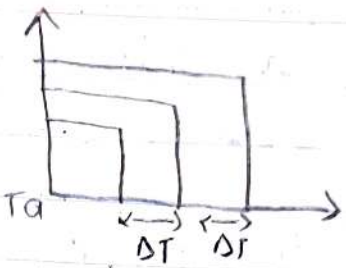
fig: process of morphing

simulating Accelerations:

- It shows the involvement of mathematics to simulate motion. As the motion may be uniform with acceleration to zero, positive or negative or non-uniform, so the combination of such motion in an animation contribute to realism.
- To import motion to a graphic object, curve fit fitting are often used for specifying the animation path betn the key frame.
- Given the vertex positions at the keyframe, we can fit the position with linear or non-linear path which determines the ~~trajections~~ trajectories for the in-betn & to simulate accelerations, we can adjust the time spacing for ~~it~~ the inbetn.

- There are different ways of simulating motion.
  ① zero acceleration (const. speed)
  ② Non-zero acceleration (quite useful)

- positive
- Negative
- combination



- for zero acceleration, the time spacing for the in betn is at equal interval.
- for positive, the time spacing betn frames should decrease & for negative, the time spacing should increase.

ΔT const.

Unit 5

Introduction to web graphics design & graphics Design Package

① Introduction to Graphic file formats

- A file format is a standard way that information is encoded for storage in a computer file.
- A file format specifies how bits are used to encode information in a digital storage medium.
- It may be either proprietary or free & may be either unpublished or open.
- for eg: video, audio, image & text file format.
- Using the right file format for the right job means the design will come out picture-perfect & just blow the intended.
- Each category has its specific use. The wrong file format could means bad print, a poor web image, a large downloaded file or missing graphic in a email.

Types of file format

① Image file format
② Video file format
③ Audio file format
④ Text file format

① Image file format:

Most image files fit into one of two general catagories i.e Rasterfile and vectorfile.

ⓐ Raster image :

They are made up of a set grid of dots called pixels. They are ~~made up~~ resolution-dependent when you stretch the pixel themselves which can result in a pixelated or blurry image.

- It is a pixel based Graphics, resolution-dependent such as photo & web graphics.

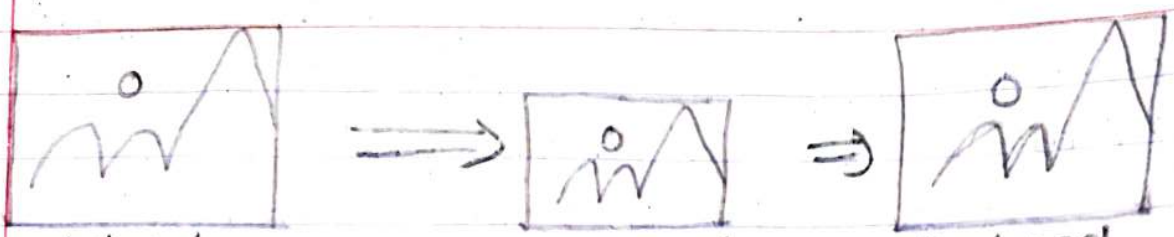
ⓑ Vector image :

They are curve-based graphics, resolution independent uses for logo. Icons & Type setting etc.

- They are digital artwork in which points, lines & curves are generated. When we shrink or enlarge a vector image, the shape of the object or image gets larger, but there won't be any detail lass or get any pixelation.

Loseless Vs Lossy file types :

- Losseless image format capture all ~~of~~ the data of the original file. Nothing from the original file, photo or piece of art is lost. Hence, the term lossless.
- The file may still be compressed, but all lossless format will be able to reconstruct the image to its original state.



Lossy Image formats what the original Image looks likes. for eg. lossy image might reduce the number of colors in image or analyze the image for any unnecessary data

- These clever technical tricks will typically reduce the file size, through they may reduce the quality of image.
- Typically, lossy file are much smaller than lossless files making them ideal online where file size & download speed are virtual.

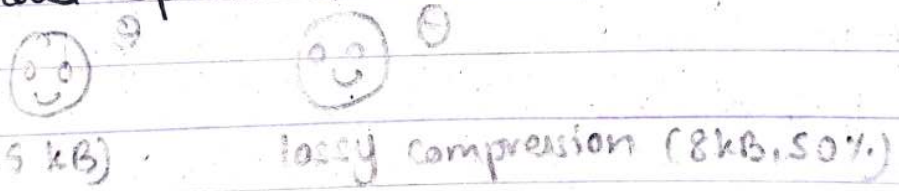


Image file format brings different ways to overcomes the problem of delivering an image with reduced file size & minimum download time.

- They are standardized means of organizationing & storing digital images.
- They may stores data in uncompressed, compressed or vector formats. Once rasterized an Image becomes a grid of pixels each of which has a no. of bits to

designate ~~the~~ its colour equal to the colour depth of the device displaying it.

- Types of image formats are GIF, JPEG, PNG etc.

① GIF : GIF stands for Graphic Interchange Format.
- Uses lossless compression technique & supports 8 bit colors.
- Supported by all browers as suitables for text, artwork, icons & cartoons.
- Large size file & its extension name is .gif.

② JPEG : JPEG stands for joint photographic Experts Group.
- Uses lossy compression technique & supports 24 bit colors.
- Supported by all browers as suitables for photograph.
- Small file size compared to GIF & its extension name is .jpg & .jpeg.

③ PNG : PNG stands for Portable Network Graphics.
- Uses ~~lossless~~ lossless compression technique & support 24 bit colon.
- Not supported by all browser so, suitable for text & icons ~~of~~ etc
- Smaller file size compared to GIF & JPEG & its extension name is .png.

Audio file format

- An audio file format is a file format for storing digital audio on a computer system.
- This data can be stored uncompressed or compressed to reduce the file size.
- It can be a raw bits stream, but it is usually a container format or an audio data format with defined storage layer.
- Types of audio format are:
  MPEG, WAVE, MIDI, WMA, AIFF & AAC etc.

(i) MPEG
- stands for moving picture Experts Group
- This files employ loss data compression.
- Digital music, audio books are often saved as mpeg's file

(ii) WAVE
- stands for Resource Interchange file format waveform
- This files employ an uncompressed audio format.
- WAVE uses .wav extension filename.

(iii) MIDI - stands for Musical Instrument Digital Interface.
- This files employ an uncompressed audio format.

(iv) WMA - stands for window Media Audio
- This files employ loss data compression developed by Microsoft.

v. AIFF

- stands for Audio integrated file format.
- This files employ uncompressed audio format.

vii AAc:

- stands for Advanced Audio coding
- This files employ loss data compression.

③ Text file format:

Files in which the bytes represent the text characters of a particular character set using specific system to relate the binary no. in the file to the text characters of the set.

- It include no. of different formatting stategies for text files in which data field are - structured in a regular pattern.

- Types of text formats are:
  ① Doc / Docx - Microsoft word software saves document
  ② .TXT - TXT documents only contain Text.
  ③ HTML - Hyper text Markup language
  ④ PDF - protable Document format
  ⑤ ZIP - compressed data files.

④ video file format:

-video file format consist of two types
  ⓐ Analog video
  ⓑ Digital video

Example of this format are as follows:

(i) MPEG :- stands for moving picture experts Group.
- This files employ loss data compression
(ii) MOV: Quick Time Movie.
(iii) WMV - window media video
(iv) AVI - Audio Video Interleave
(v) Real video - proprietary file format
(vi) SWF: shook wave flash means small web format

Graphics package.

A graphics package is an application software that can be used to create & manipulate image on a computer. For eg: graphics package include Ms point, Adobe photoshop, Instant Artist, corel Draw etc.

There are two main types of graphic package.

① painting package
- It produces images by changing the color of pixel on the screen.
- These are coded as a pattern of bits to create a bitmapped graphics file.
- Bitmapped graphics are used for image such as scanned photographs or pictures taken with a digital camera.

② Drawing package :-
- A drawing package produce images that are made up

from colored lines & shapes such as circles, squares & rectangle.
- when an image is saved it is stored in a vector graphics files such as a series of instructions which can be used to recreate it.

Advantages of painting package.
- The main advantage offered by this type of graphics is that individual pixel can be changed which makes very detailed editing possible.

Disadvantages of painting package
- Individual parts of an image cannot be resized.
- Only the whole picture can be increased or decreased in size.
- Information has to be stored about ~~one~~ every pixel in an image ~~include~~ which produces files that use large amount of backing storage space.
- Eg: graphics package that produce bitmapped image include: ms-point, pc paint brush, Adobe photoshop & JASC's point shop pro. etc.

Advantages of Drawing package
- They use less storage space than bitmap graphics
- Each part of an image is treated as a separate object, which means that individual parts can be easily modified.

Disadvantages of Drawing package.

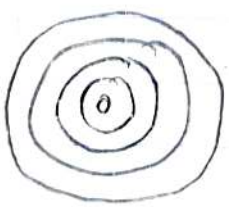
- They don't look at realistic as bitmap graphics.
- Example of drawing graphic's package include corel draw, micrographix Designer & computer aided Design C(AD) package such as Auto CAD.

Basic principle of Graphic Design

① proximity:

The grouping & shaping of object on page.

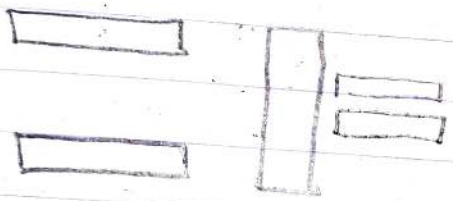
- Items relating to each other should be grouped close together.
- when several items are in close proximity to each other, they become one visual unit rather than several separate units.
- This help organize information & reduces clutter.
- Items that are not related to each other should not be in close proximity.
- The closeness or lack of closeness indicates the relationships.
- Elements that are intellectually connected should be visually connected.



② Alignment:

Keeping object in line with one another.

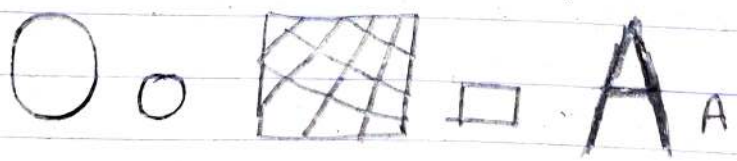
- Nothing should be placed on the page arbitarily.
- Every element should have some visual connection with another element on the page.
- This creates a clean, sophisticated look.
- Arrangements of all edges with equal space is done.



③ Contrast:

: Creating distinction by highlighting differences.

- Avoid elements on the pages that are merely similar.
- If the elements (type, color, size, line, thickness, shape, space etc.) are not the same than make them very different.
- Contrast is often the most important visual elements on the pages.



④ Repition

- Repeat visual element of the design throughout the piece. you can repeat color, shape texture, spatial, relationships, line, thick-ness, size etc.
- This helps develop the organization & strengthens the

unity.

What goes Around
comes Around

⑤ White space.

uses of Graphic package

- There are numerous uses of Graphic package depending on the type of graphics you are creating.
- It can be used to scan documents & images into the computer system.
- It is used to edit Images.
- It can be used to create brand identity such as logo, business cards etc.
- It can be used to create a print medium for digital display or printing such as books, magazines etc.
- It can be used to create online & offline marketing materials such as flyers, banners etc.
- some package can be used to create visual material for product differentiation. For eg: product label, packaging design etc.
- It can be used to create communication patterns

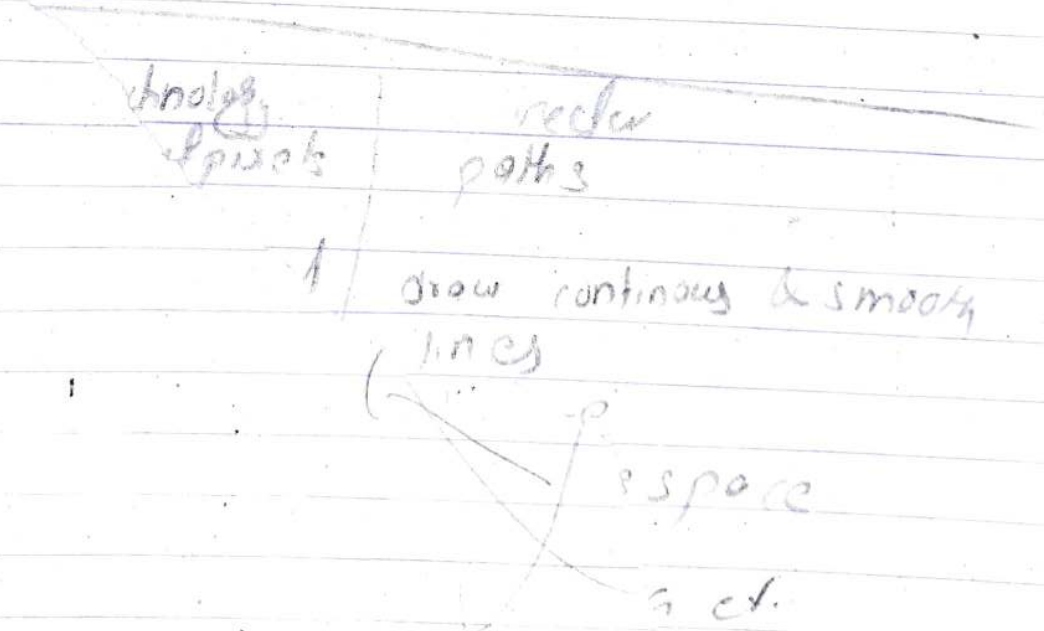
For eg: T-shirt design, create concept arts

- It can be used to create environment using computer aided visual patterns. For eg: street sign.
- It can be used to teach element. known as computer aided modelling
  students.
- It can be used to view in

The purpose of CAM is using 3D models to design machining processes

software is mostly used by a trained machinist

example: power MILL, Solid CAM



CAD	CAM
- stands for computer Aided Design	computer aided manufacturing
- It is also known as computer Aided Drafting	known as computer Aided modelling
- The purpose of CAD is making 2D technical drawings & 3D models	The purpose of CAM is using 3D models to design machining processes
- Software is mostly used by an engineer	Software is mostly used by a trained machinist.
- example: autoCAD, Autodesk etc	example: power MILL, Solid CAM

Raster display Technology	vector
They are composed of pixels	paths
- used to draw mathematical curves, polygons etc	draw continous & smooth lines
- cost less	cost more
- occupy more space	occupy less space
- eg: JPEG, GIF, BMP	eg. EPS, SVG etc.