

Computer Graphics

Prof. Vijay Bagdi

Q. Write Application of Computer Graphics

Graphics Ans:

Application of Computer Graphics

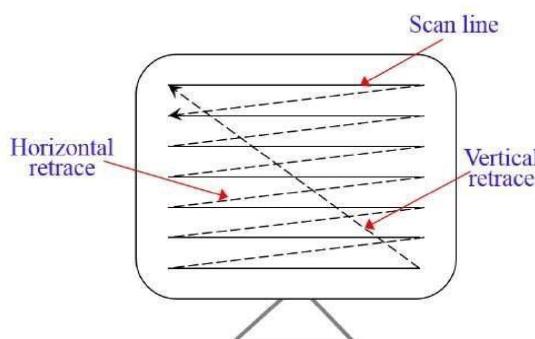
Computer Graphics has numerous applications, some of which are listed below –

- **Computer graphics user interfaces (GUIs)** – A graphic, mouse-oriented paradigm which allows the user to interact with a computer.
- **Business presentation graphics** – "A picture is worth a thousand words".
- **Cartography** – Drawing maps.
- **Weather Maps** – Real-time mapping, symbolic representations.
- **Satellite Imaging** – Geodesic images.
- **Photo Enhancement** – Sharpening blurred photos.
- **Medical imaging** – MRIs, CAT scans, etc. - Non-invasive internal examination.
- **Engineering drawings** – mechanical, electrical, civil, etc. - Replacing the blueprints of the past.
- **Typography** – The use of character images in publishing - replacing the hard type of the past.
- **Architecture** – Construction plans, exterior sketches - replacing the blueprints and hand drawings of the past.
- **Art** – Computers provide a new medium for artists.
- **Training** – Flight simulators, computer aided instruction, etc.
- **Entertainment** – Movies and games.
- **Simulation and modeling** – Replacing physical modeling and enactments

Q. What is Raster scan display in computer

graphics? Ans:

- 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 moves across each row, the beam intensity is turned on and off to create a pattern of illuminated spots.
- Picture definition is stored in memory area called the Refresh Buffer or Frame 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 (scan line) at a time as shown in the following illustration Each screen point is referred to as a **pixel (picture element) or pel**.
- At the end of each scan line, the electron beam returns to the left side of the screen to begin displaying the next scanline



Raster-scan system:-

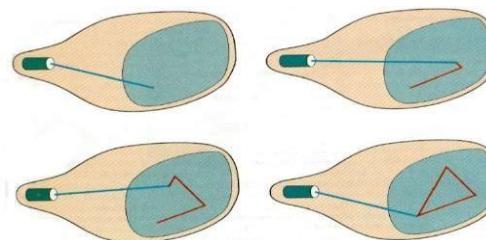
1. Raster displays have less resolution.
2. The lines produced are ziz-zag as the plotted values are discrete.
3. High degree realism is achieved in picture with the aid of advanced shading and hidden surface technique.
4. Decreasing memory costs have made raster systems popular.
5. In this case, the electron beam is swept across the screen, one row at a time from top to bottom.
6. Picture definition is stored in a memory area called the refresh buffer/frame buffer.
7. Refreshing on raster scan displays is carried out at the rate of 60 to 80 frames/second.

Q. What is Random Scan display in computer

graphics? Ans:

- In this technique, the electron beam is directed only to the part of the screen where the picture is to be drawn rather than scanning from left to right and top to bottom as in raster scan.
- It is also called vector display, stroke-writing display, or calligraphic display.
- Picture definition is stored as a set of line-drawing commands in an area of memory referred to as the **refresh display file**.

- To display a specified picture, the system cycles through the set of commands in the display file, drawing each component line in turn.
- After all the line-drawing commands are processed, displays are designed to draw all the component lines of a picture 30 to 60 times each second. Processed, the system cycles back to the first line command in the list.



Random scan system:-

- Random displays have high resolutions since the picture definition is stored as a set of line drawing commands and not as a set of intensity values.
- Smooth lines are produced as the electron beam directly follows the line path.
- Realism is difficult to achieve.
- Random-scan systems are generally costlier.
- Here CRT has the electron beam directly only to the parts of the screen where a picture is to be drawn.
- Picture definition is stored as a set of line drawing commands in an area of memory referred to as refresh display file.
- Random scan systems are designed to draw all the component lines of a picture 30 to 60 times each second. Dig shown below

Q. Differences between raster and random scan

Ans:

Sr. No.	Raster scan	Sr. No.	Random Scan
1	Raster scan system has less resolution	1	Random display has higher resolution
2	The lines produced in raster scan are zigzag	2	The lines produced in it are smooth as electron beam directly follows the line path
3	Decreasing memory cost in raster scan has made it popular	3	Random Scan is Costlier
4	Electron beam is swept across the screen, one row at a time from top to bottom in case of raster scan	4	Random scan electron beam directly to those parts of the screen where picture needs to be drawn
5	While refreshing on raster scan is carried out at a rate of 60-80 frames per second.	5	Random scan draws component lines at the rate of 30-60 times each second

Q. Explain types of Computer Graphics.

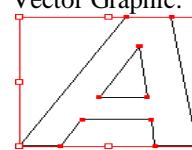
Ans:

- There are two kinds of computer graphics - raster (composed of pixels) and vector (composed of paths). Raster images are more commonly called *bitmap* images.
- A bitmap image uses a grid of individual pixels where each pixel can be a different color or shade. Bitmaps are composed of pixels.
- Vector graphics use mathematical relationships between points and the paths connecting them to describe an image. Vector graphics are composed of paths.
- The image to the left below is representative of a bitmap and the image to the right is representative of a vector graphic. They are shown at four times of actual size to exaggerate the fact that the edges of a bitmap become jagged as it is scaled up:

Bitmap Image:



Vector Graphic:



- The jagged appearance of bitmap images can be partially overcome with the use of "anti-aliasing".
- Anti-aliasing is the application of subtle transitions in the pixels along the edges of images to minimize the jagged effect (below left). A scalable vector image will always appear smooth (below right).

Anti-Aliased Bitmap Image:



Smooth Vector Image:



- Bitmap images require higher resolutions and anti-aliasing for a smooth appearance.
- Vector-based graphics on the other hand are mathematically described and appear smooth at any size or resolution.
- Bitmaps are best used for photographs and images with subtle shading.
- Graphics best suited for the vector format are page layout, type, line art or illustrations

Q. Describe graphics pipeline in hardware

Ans:

- The rendering pipeline is mapped onto current graphics acceleration hardware such that the input to the graphics card (GPU) is in the form of vertices.
- These vertices then undergo transformation and per-vertex lighting.
- At this point in modern GPU pipelines a custom vertex shader program can be used to manipulate the 3D vertices prior to rasterization.
- Once transformed and lit, the vertices undergo clipping and rasterization resulting in fragments.
- A second custom shader program can then be run on each fragment before the final pixel values are output to the frame buffer for display.
- The graphics pipeline is well suited to the rendering process because it allows the GPU to function as a stream processor since all vertices and fragments can be thought of as independent.
- This allows all stages of the pipeline to be used simultaneously for different vertices or fragments as they work their way through the pipe.
- In addition to pipelining vertices and fragments, their independence allows graphics processors to use parallel processing units to process multiple vertices or fragments in a single stage of the pipeline at the same time

Q. Write note on Image Scanner.

Ans:

- A scanner is a device that optically scans images, printed text, handwriting, or an object, and converts it to a digital image. Common examples found in offices are variations of the desktop (or flatbed) scanner where the document is placed on a glass window for scanning.
- Hand-held scanners, where the device is moved by hand, have evolved from text scanning "wands" to 3D scanners used for industrial design, reverse engineering, test and measurement, orthotics, gaming and other applications. Mechanically driven scanners that move the document are typically used for large-format documents, where a flatbed design would be impractical.
- Modern scanners typically use a charge-coupled device (CCD) or a Contact Image Sensor (CIS) as the image sensor, whereas older drum scanners use a photomultiplier tube as the image sensor. A rotary scanner, used for high-speed document scanning, is another type of drum scanner, using a CCD array instead of a photomultiplier.
- Other types of scanners are planetary scanners, which take photographs of books and documents, and 3D scanners, for producing three-dimensional models of objects.
- Another category of scanner is digital camera scanners, which are based on the concept of reprographic cameras.
- Due to increasing resolution and new features such as anti-shake, digital cameras have become an attractive alternative to regular scanners.
- While still having disadvantages compared to traditional scanners (such as distortion, reflections, shadows, low contrast), digital cameras offer advantages such as speed, portability, gentle digitizing of thick documents without damaging the book spine.
- New scanning technologies are combining 3D scanners with digital cameras to create full-color, photo-realistic 3D models of objects.

Q. What is raster image? Enlist components of Raster system.

Ans:

- A raster image is a collection of dots called pixels.
- Each pixel is tiny Colored Square. When an image is scanned, the image is converted to a collection of pixels called a raster image.
- Scanned graphics & web graphics (jpg and GIF) are the most common forms of raster images.
 - Raster Scan display components: For raster scan display following are the three components.
 - The frame Buffer which is also the refresh buffer or bitmap.

- It is the refresh storage area in the digital memory, in which the matrix or array of intensity values and other parameters called attribute of all the pixels making up the image are stored in binary form.
- The display device which converts the electrical signals into visible images namely the VDU.
- The display controller, the interface that transmits the contents of the frame buffer to the VDU is a form compatible with display device, a certain no of (30 or more) times a second.
- Display controllers also generate line segment and text character.
- Creating points, lines, and characters as well as filling its areas with shades or colours are all accomplished by this scan technique called a FRAME BUFFER DISPLAY.

A common method for storing character is to store the pixel information.

1. The sequence of operation is as follows:
2. Graphics Command
3. Display Processor (scan conversion)
4. Frame Buffer
5. Display Controller
6. VDU

Q. What is Video display controller?

Ans:

- A video display controller or VDC is an integrated circuit which is the main component in a video signal generator, a device responsible for the production of a TV video signal in a computing or game system.
- Some VDCs also generate an audio signal, but in that case it is not their main function.
- VDCs were most often used in the old home-computers of the 80s, but also in some early video game systems.
- The VDC is always the main component of the video signal generator logic, but sometimes there are also other supporting chips used, such as RAM to hold the pixel data, ROM to hold character fonts, or perhaps some discrete logic such as shift registers were necessary to build a complete system.
- In any case, it's the VDC's responsibility to generate the timing of the necessary video signals, such as the horizontal and vertical synchronization signals, and the blanking interval signal.

Q. Write Short note on:

Ans:

- i. Calligraphic Refresh graphics display
- ii. Raster refresh graphics display
- iii. Pixels and Frame buffer

Calligraphic Refresh graphics display

- A Calligraphic (line drawing or vector) refresh CRT display uses a very short persistence phosphor.
- These displays are frequently called random scan display.
- Because of the short persistence of the phosphor, the picture painted on the CRT must be repainted or refreshed many times each second.
- The minimum refresh rate is at least 30 times each second. Refresh rates much lower than 30 times each second result in a flickering image.
- The basic calligraphic refresh display requires two elements in addition to the CRT. These are the display buffer and the display -controller.
- The display buffer is contiguous memory containing all the information required to draw the picture on the CRT. The display controller's function is to repeatedly cycle through this information at the refresh rate.
- Two factors which limit the complexity (number of vectors displayed) of the picture are the size of the display buffer and the speed of the display controller.
- A further limitation is the speed at which picture information can be processed.
- Features of calligraphic refresh displays
 - It is a vector graphics display.
 - Resolution is the same as storage tube display.
 - Employee the concept of picture segmentation that support the interactive graphics programs

Raster refresh graphics display

- A raster CRT graphics device can be considered a matrix of discrete cells each of which can be made bright.
- Thus it is a point plotting devices. It is not possible except in special cases to directly draw a straight line from one addressable point, or pixel in the matrix to another addressable point, or pixel.

- The line can only be approximated by a series of dots (pixels) close to the path of the line. Only in the special cases of completely horizontal, vertical or 45 degree lines will a straight line result.
- All other lines will appear as a series of stair steps. This is called aliasing. The most common method of implementing a raster CRT graphics device utilizes a frame buffer. A frame buffer is a large, contiguous piece of computer memory. As a minimum there is one memory bit for each location or pixel in the raster. This amount of memory is called a bit plane. A 512 X 512 element square raster requires 2^{18} memory bits in a single bit plane. The picture is built up in the frame buffer 1 bit at a time.

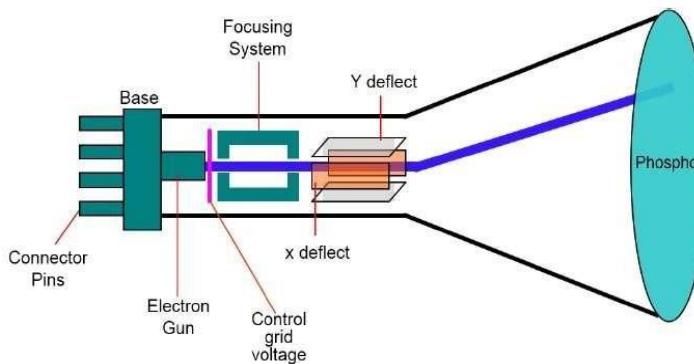
Pixels and Frame buffer

- We cannot represent an infinite number of points (pixels) on a computer, just as we cannot do that with numbers.
- The machine is finite, and we are limited to a finite number of points making up each line. The maximum number of points (pixels) which a line can have is a measure of the resolution of the display device. The greater the number of points, the higher the resolution. Resolution is the number of visibly distinct dots that can be displayed in a given area of the screen.
- Computer graphics is an art of drawing pictures on computer screens with the help of programming. It involves computations, creation, and manipulation of data. In other words, we can say that computer graphics is a rendering tool for the generation and manipulation of images.

Q. Explain Cathode Ray Tube

Ans:

- The primary output device in a graphical system is the video monitor. The main element of a video monitor is the Cathode Ray Tube (CRT), shown in the following illustration.



- The operation of CRT is very simple –
 - The electron gun emits a beam of electrons (cathode rays).
 - The electron beam passes through focusing and deflection systems that direct it towards specified positions on the phosphor-coated screen.
 - When the beam hits the screen, the phosphor emits a small spot of light at each position contacted by the electron beam.

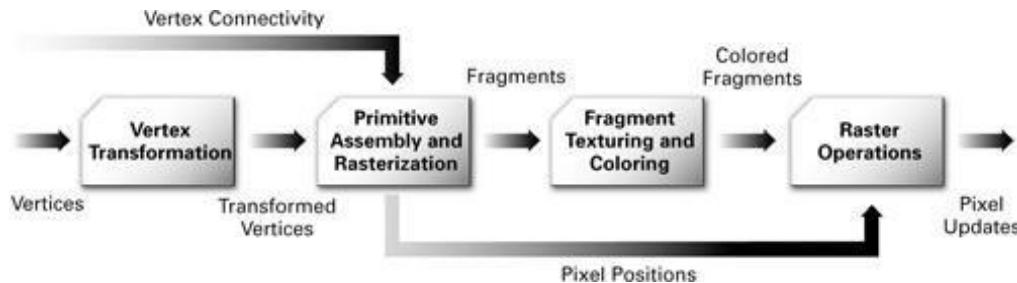
- It redraws the picture by directing the electron beam back over the same screen points quickly.
- There are two ways (Random scan and Raster scan) by which we can display an object on the screen.

Q. Explain the graphics pipeline.

Ans:

Graphics pipeline

- The *graphics pipeline* is a fundamental concept in computer graphics and refers to a series of interconnected stages through which data and commands describing a scene go through when being rendered.
- The term *rendering pipeline* is also used, sometimes synonymously, but often to refer to non-rasterisation based rendering approaches.



- In 3D computer graphics, the graphics pipeline or rendering pipeline refers to the sequence of steps used to create a 2D raster representation of a 3D scene.

Stages of the graphics pipeline

- Geometric primitives**

First, the scene is created out of geometric primitives. Traditionally this is done using triangles, which are particularly well suited to this as they always exist on a single plane.

- **Modeling and transformation**

Transform from the local coordinate system to the 3d world coordinate system. A model of a teapot in abstract is placed in the coordinate system of the 3d world.

- **Camera transformation**

Transform the 3d world coordinate system into the 3d camera coordinate system, with the camera as the origin. **Lighting** illuminate according to lighting and reflectance. If the teapot is a brilliant white color, but in a totally black room, then the camera sees it as black. In this step the effect of lighting and reflections are calculated.

- **Projection transformation**

Transform the 3d world coordinates into the 2d view of the camera, for instance the object the camera is centered on would be in the center of the 2d view of the camera.^[3] In the case of a Perspective projection, objects which are distant from the camera are made smaller. This is achieved by dividing the X and Y coordinates of each vertex of each primitive by its Z coordinate(which represents its distance from the camera). In an orthographic projection, objects retain their original size regardless of distance from the camera.

- **Clipping**

Geometric primitives that now fall completely outside of the viewing frustum will not be visible and are discarded at this stage.

- **Scan conversion or rasterization**

Rasterization is the process by which the 2D image space representation of the scene is converted into raster format and the correct resulting pixel values are determined. From now on, operations will be carried out on each single pixel. This stage is rather complex, involving multiple steps often referred as a group under the name of pixel pipeline.

- **Texturing, fragment shading**

At this stage of the pipeline individual fragments (or pre-pixels) are assigned a color based on values interpolated from the vertices during rasterization, from a texture in memory, or from a shader program.

Q . Explain Hardcopy Technology.

Ans:

- A hard copy is a printed document. It may be a text file, photograph, drawing, or any other type of printable file. For example, instead of e-mailing a business memo, it may be sent out as a hard copy, or an actual physical paper containing the memo
- When a document is created on a computer, it is typically saved as a file on the the computer's hard drive. This is sometimes referred to as a soft copy.
- While the file can be easily opened and edited on a computer, it can also be easily deleted.
- Therefore, sometimes printing a hard copy is done to create a physical backup of the document.

Printer

- A printer is an output device that prints paper documents.
- This includes text documents, images, or a combination of both. The two most common types of printers are inkjet and laser printers. Inkjet printers are commonly used by consumers, while laser printers are a typical choice for businesses.
- Dot matrix printers, which have become increasingly rare, are still used for basic text printing.
- The printed output produced by a printer is often called a hard copy, which is the physical version of an electronic document.
- While some printers can only print black and white hard copies, most printers today can produce color prints.
- In fact, many home printers can now produce high-quality photo prints that rival professionally developed photos. This is because modern printers have a high DPI (dots per inch) setting, which allows documents to be printed with a very fine resolution.

Plotter:

- Plotter is a printer designed for printing vector graphics. Instead of printing individual dots on the paper, plotters draw continuous lines. This makes plotters ideal for printing architectural blueprints, engineering designs, and other CAD drawings.
- There are two main types of plotters – drum and flatbed plotters. Drum plotters (also called roller plotters) spin the paper back and forth on a cylindrical drum while the ink pens move left and right.
- By combining these two directions, lines can be drawn in any direction. Flatbed plotters have a large horizontal surface on which the paper is placed.
- A traveling bar draws lines on the paper as it moves across the surface.

Q. Explain Homogenous Coordinates.

Ans:**Homogenous Coordinates**

- To perform a sequence of transformation such as translation followed by rotation and scaling, we need to follow a sequential process :
 - Translate the co-ordinates, Rotate the translated coordinates, and then Scale the rotated coordinates to complete the composite transformation.
 - To shorten this process, we have to use 3×3 transformation matrix instead of 2×2 transformation matrix. To convert a 2×2 matrix to 3×3 matrix, we have to add an extra dummy coordinate W.
 - In this way, we can represent the point by 3 numbers instead of 2 numbers, which is called Homogenous Coordinate system. In this system, we can represent all the transformation equations in matrix multiplication.
 - Any Cartesian point $P(X, Y)$ can be converted to homogenous coordinates by $P' (X_h, Y_h, h)$.

Q. Explain Transformation and different types.**Ans:**

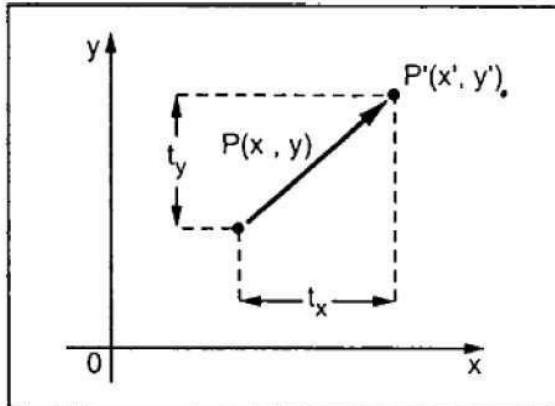
- Transformation mainly refers to the alteration of picture.
- This method allows the uniform alteration of entire picture.
- Transformation means changing some graphics into something else by applying rules.
- We can have various types of transformations such as translation, scaling up or down, rotation, shearing, etc.
- When a transformation takes place on a 2D plane, it is called 2D transformation.
- Transformations play an important role in computer graphics to reposition the graphics on the screen and change their size or orientation.

There are following types of transformation

- Translation
- Scaling
- Rotation
- Reflection
- Shear

Translation

- A translation moves an object to a different position on the screen. You can translate a point in 2D by adding translation coordinate (t_x, t_y) to the original coordinate (X, Y) to get the new coordinate (X', Y') .



- From the above figure, you can write that –

$$X' = X + T_x$$

$$Y' = Y + T_y$$

Where (T_x, T_y) is called the translation factors.

We can write it as – $P' = P + T$

- Translation Matrix is**

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{bmatrix}$$

$$[x' \ y' 1] = [x \ y 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{bmatrix}$$

Scaling

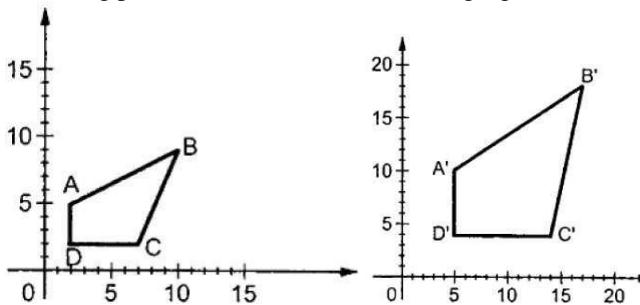
- To change the size of an object, scaling transformation is used.
- In the scaling process, you either expand or compress the dimensions of the object. Scaling can be achieved by multiplying the original coordinates of the object with the scaling factor to get the desired result.
- Let us assume that the original coordinates are (X, Y) , the scaling factors are (S_x, S_y) , and the produced coordinates are (X', Y') .
- This can be mathematically represented as shown below –
 $X' = X \cdot S_x$ and
 $Y' = Y \cdot S_y$

The scaling factor S_x, S_y scales the object in X and Y direction respectively. The above equations can also be represented in matrix form as below –

$$\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$[x' \ y' 1] = [x \ y 1] \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

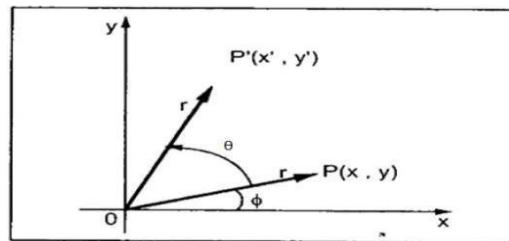
The scaling process is shown in the following figure.



- If we provide values less than 1 to the scaling factor S, then we can reduce the size of the object. If we provide values greater than 1, then we can increase the size of the object.

Rotation

- In rotation, we rotate the object at particular angle θ (theta) from its origin.
- From the following figure, we can see that the point P(X, Y) is located at angle ϕ from the horizontal X coordinate with distance r from the origin.
- Let us suppose you want to rotate it at the angle θ . After rotating it to a new location, you will get a new point P' (X', Y').



Rotation: For clockwise direction

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$[x' \ y' 1] = [x \ y 1] \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation: For anticlockwise direction

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$[x' \ y' 1] = [x \ y 1] \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Reflection

- Reflection is the mirror image of original object. In other words, we can say that it is a rotation operation with 180° .
- In reflection transformation, the size of the object does not change.
- This type of transformation allows obtaining the mirror image of an object.
- To carry out the reflection it is necessary to define the axis of reflection.

The Transformation matrix for the axis as refection is,

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The transformation matrix for the y axis reflection is,

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The Transformation matrix for refaction is, $y = -x$ as

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The Transformation matrix for the $y = x$ axis as reflection is,

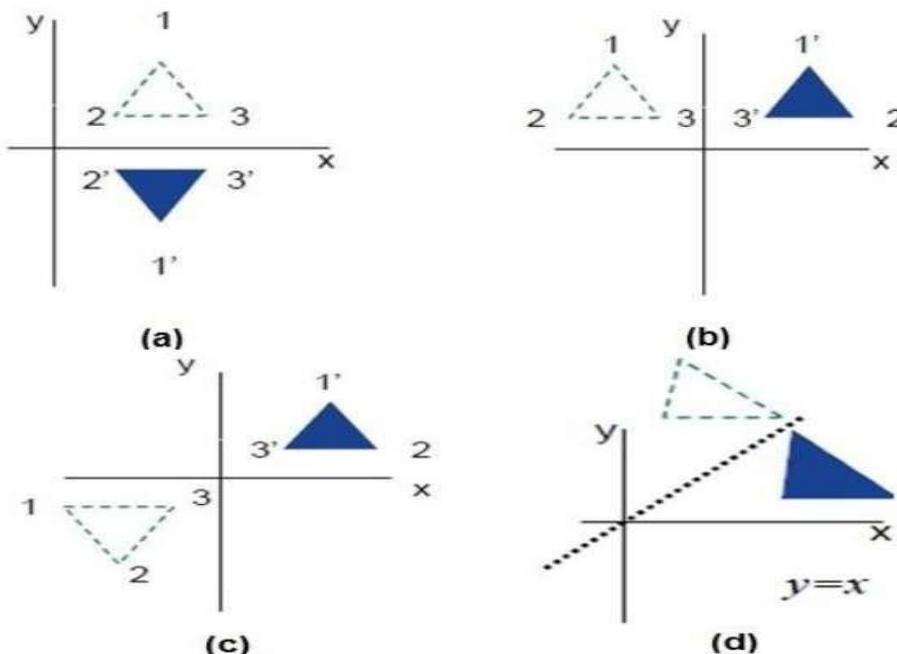
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Reflection about arbitrary axis:

- If the reflection is to be made about an axis which is not the standard one such as $y = x$ etc. then this axis must be first coincide with one of the standard axis and then the reflection is formed.

The steps are as follows:

- Translation: Firstly translate the line to the origin
- Rotation: If needed then rotate it to coincide it with one of the standard axis.
- Reflection: Take reflection.
- Re-Rotation: If the line is rotated in step(ii) rerotation is performed in reverse direction.
- Re-Translation: Retranslate the line back to its original position.

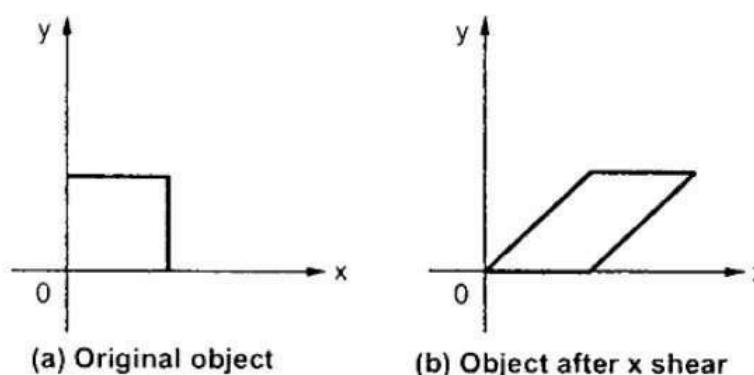


Shear

- A transformation that slants the shape of an object is called the shear transformation.
- There are two shear transformations X-Shear and Y-Shear.
- One shift X co-ordinates values and other shifts Y coordinate values.
- However; in both the cases only one coordinate changes its coordinates and other preserves its values.
- Shearing is also termed as Skewing.

X-Shear

The X-Shear preserves the Y coordinate and changes are made to X coordinates, which causes the vertical lines to tilt right or left as shown in below figure.

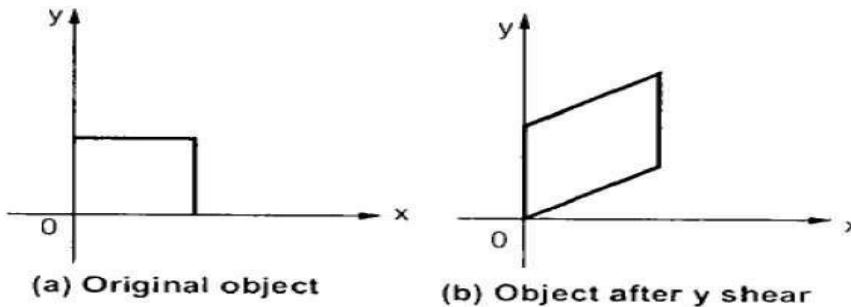


The transformation matrix for X-Shear can be represented as –

$$\text{X- Shear} \begin{bmatrix} 1 & 0 & 0 \\ Shx & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Y-Shear

The Y-Shear preserves the X coordinates and changes the Y coordinates which causes the horizontal lines to transform into lines which slopes up or down as shown in the following figure.



The Y-Shear can be represented in matrix form as –

$$\text{Y shear} \begin{bmatrix} 1 & Shy & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Shx and Shy are referred as shearing parameters or shearing factors.

Combined Transformation

- To control the size of an object, we need to perform matrix operations on the position vector which defines the vertices.
- It is not necessary that we get the required orientation by applying single transformation; it may require more than one transformation.
- Since matrix multiplication is non commutative, the order of application of the transformation is important.

We can set up a matrix for any sequence of transformation as a composite transformation matrix by calculating the matrix product of the individual transformation:

- i. If two successive translation are applied then

$$\begin{pmatrix} 1 & 0 & tx1 \\ 0 & 1 & ty1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & tx2 \\ 0 & 1 & ty2 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & tx1 + tx2 \\ 0 & 1 & ty1 + ty2 \\ 0 & 0 & 1 \end{pmatrix}$$

- ii. If two successive rotations θ_1 and θ_2 then

$$\begin{pmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- iii. If two Successive scaling are applied then

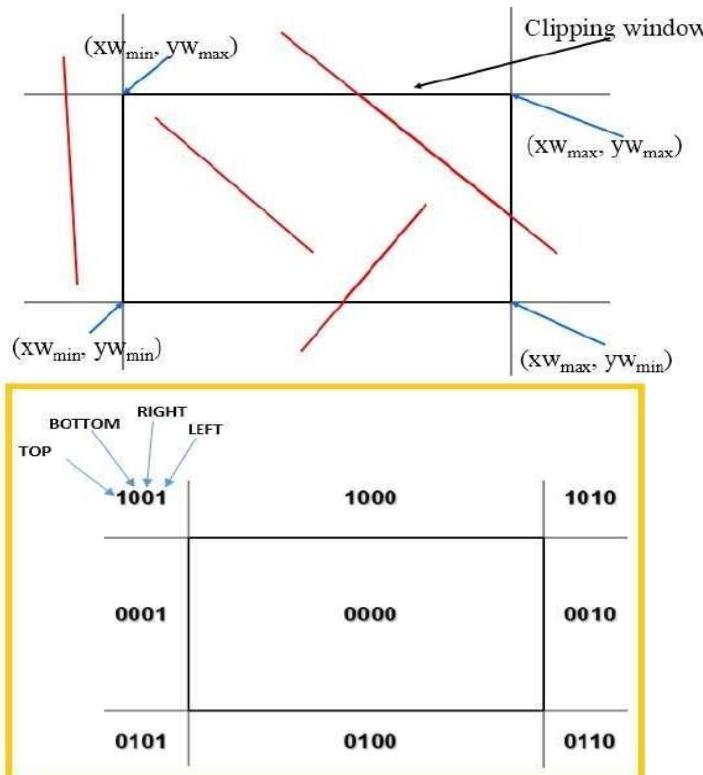
$$\begin{pmatrix} Sx1.Sx2 & 0 & 0 \\ 0 & Sy1.Sy2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

2D Clipping Algorithm

Q. Explain Sutherland Cohen code Algorithm.

Ans:

This algorithm uses the clipping window as shown in the following figure



The minimum coordinate for the clipping region is (XW_{min}, YW_{min}) and the maximum coordinate for the clipping region is (XW_{max}, YW_{max}).

We will use 4-bits to divide the entire region. These 4 bits represent the Top, Bottom, Right, and Left of the region as shown in the following figure.

Here, the TOP and LEFT bit is set to 1 because it is the TOP-LEFT corner.

There are 3 possibilities for the line –

- Line can be completely inside the window (This line will be completely removed from the region).
- Line can be completely outside of the window (This line will be completely removed from the region).
- Line can be partially inside the window (We will find intersection point and draw only that portion of line that is inside region).

Algorithm

Step 1 – Assign a region code for each endpoints.

Step 2 – If both endpoints have a region code 0000 then accept this line.

Step 3 – Else, perform the logical AND operation for both region codes.

Step 3.1 – If the result is not 0000, then reject the line.

Step 3.2 – Else you need clipping.

Step 3.2.1 – Choose an endpoint of the line that is outside the window.

Step 3.2.2 – Find the intersection point at the window boundary (base on region code).

Step 3.2.3 – Replace endpoint with the intersection point and update the region code.

Step 3.2.4 – Repeat step 2 until we find a clipped line either trivially accepted or trivially rejected.

Step 4 – Repeat step 1 for other lines.

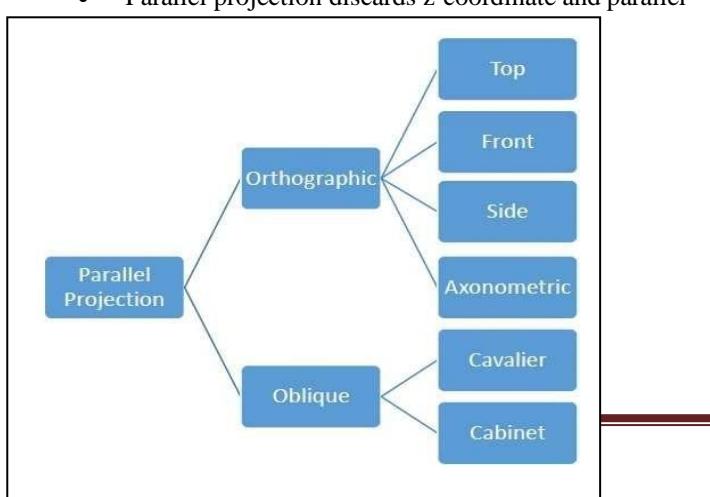
Q. Explain Parallel Projection.

Ans:

- Parallel projection discards z-coordinate and parallel

lines from each vertex on the object are extended until they intersect the view plane. In parallel projection, we specify a direction of projection instead of center of projection.

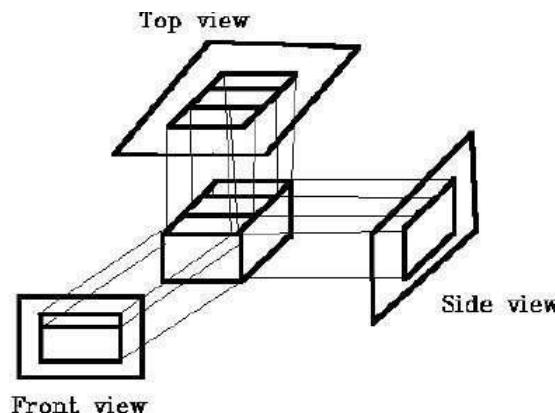
- In parallel projection, the distance from the center of projection to project plane is infinite. In this type of projection, we connect the projected vertices by line segments which correspond to connections on the original object.
- Parallel projections are less realistic, but they are good for exact measurements. In this type of



Q. Explain Orthographic Projection.

Ans:

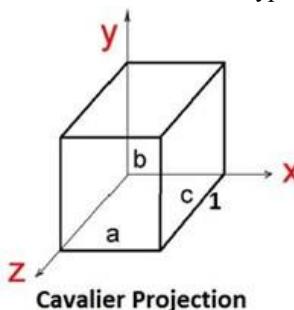
- In orthographic projection the direction of projection is normal to the projection of the plane.
- There are three types of orthographic projections –
 1. Front Projection
 2. Top Projection
 3. Side Projection



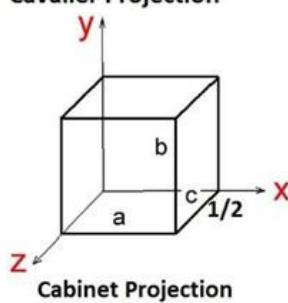
Q. Explain Oblique Projection.

Ans:

- In orthographic projection, the direction of projection is not normal to the projection of plane.
- In oblique projection, we can view the object better than orthographic projection.
- There are two types of oblique projections – Cavalier and Cabinet.



The Cavalier projection makes 45° angle with the projection plane. The projection of a line perpendicular to the view plane has the same length as the line itself in Cavalier projection. In a cavalier projection, the foreshortening factors for all three principal directions are equal.

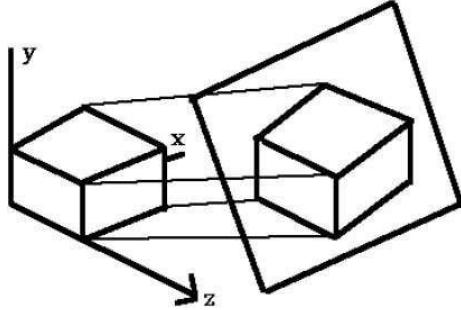


The Cabinet projection makes 63.4° angle with the projection plane. In Cabinet projection, lines perpendicular to the viewing surface are projected at $\frac{1}{2}$ their actual length. Both the projections are shown in the following figure –

Q. Explain Isometric Projections.

Ans:

- Orthographic projections that show more than one side of an object are called axonometric orthographic projections.

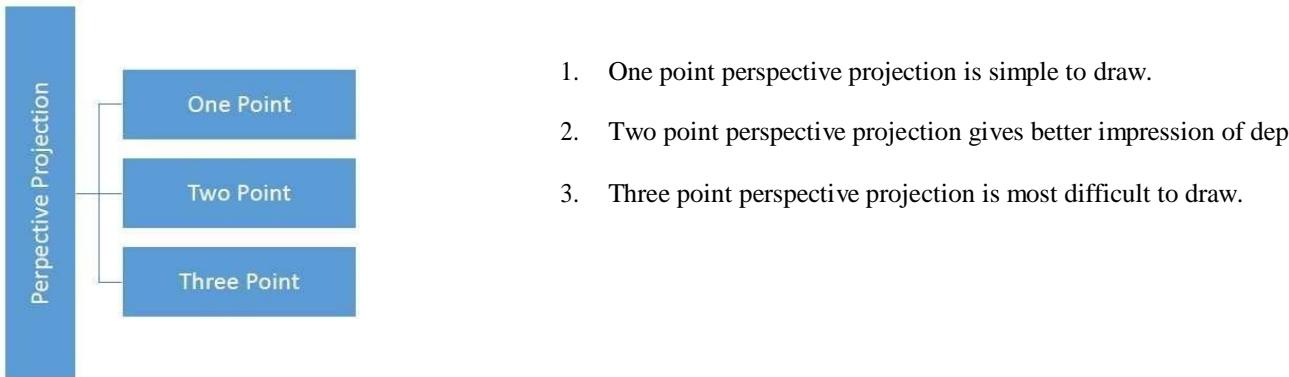


- The most common axonometric projection is an isometric projection where the projection plane intersects each coordinate axis in the model coordinate system at an equal distance.
- In this projection parallelism of lines are preserved but angles are not preserved. The following figure shows isometric projection –

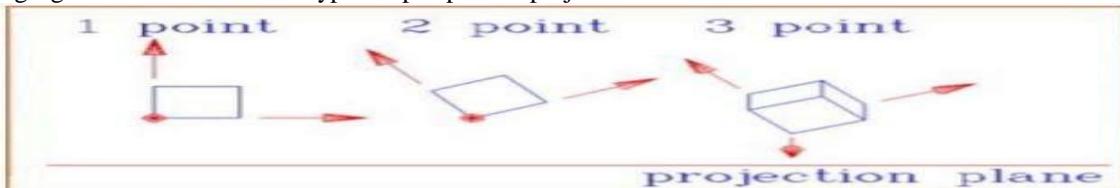
Q. Explain Perspective Projection.

Ans:

- In perspective projection, the distance from the center of projection to project plane is finite and the size of the object varies inversely with distance which looks more realistic.
- The distance and angles are not preserved and parallel lines do not remain parallel. Instead, they all converge at a single point called center of projection or projection reference point.
- There are 3 types of perspective projections which are shown in the following chart.



The following figure shows all the three types of perspective projection –



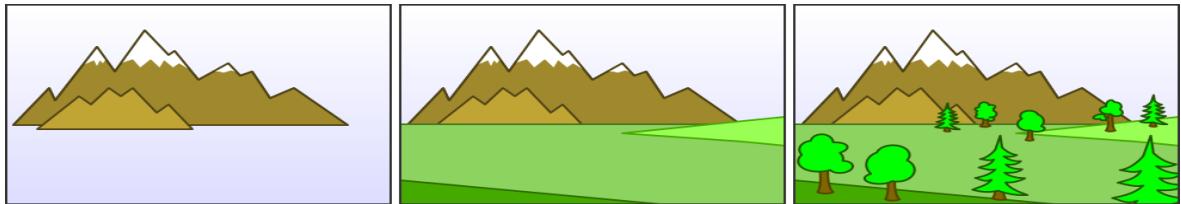
- When we view a picture containing non-transparent objects and surfaces, then we cannot see those objects from views which are behind from objects closer to eye.
- We must remove these hidden surfaces to get a realistic screen image. The identification and removal of these surfaces is called Hidden-surface problem.
- There are two approaches for removing hidden surface problems – Object-Space method and Image-space method.
- The Object-space method is implemented in physical coordinate system and image-space method is implemented in screen coordinate system.
- When we want to display a 3D object on a 2D screen, we need to identify those parts of a screen that are visible from a chosen viewing position.

Q. What is Painter's Algorithm.

Ans:

- The painter's algorithm, also known as a priority fill, is one of the simplest solutions to the visibility problem in 3D computer graphics.
- When projecting a 3D scene onto a 2D plane, it is necessary at some point to decide which polygons are visible, and which are hidden.
- The name "painter's algorithm" refers to the technique employed by many painters of painting distant parts of a scene before parts which are nearer thereby covering some areas of distant parts.
- The painter's algorithm sorts all the polygons in a scene by their depth and then paints them in this order, farthest to closest. It will paint over the parts that are normally not visible — thus solving the visibility problem — at the cost of having painted invisible areas of distant objects.

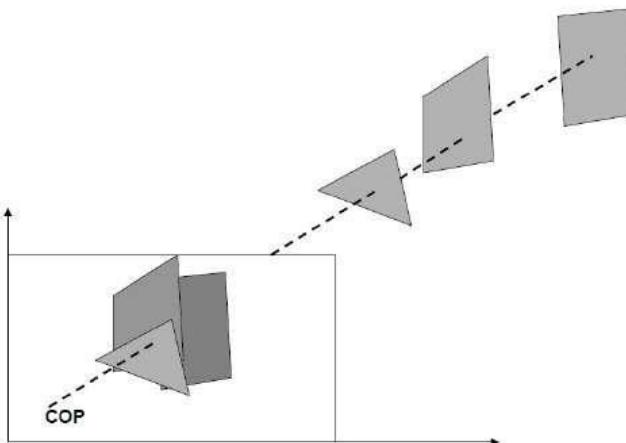
- The ordering used by the algorithm is called a '*depth order*', and does not have to respect the numerical distances to the parts of the scene: the essential property of this ordering is, rather, that if one object obscures part of another then the first object is painted after the object that it obscures.
- Thus, a valid ordering can be described as a topological ordering of a directed acyclic graph representing occlusions between objects



Q. Explain Depth Buffer (Z-Buffer) Method.

Ans:

- This method is developed by Cut mull.
- It is an image-space approach. The basic idea is to test the Z-depth of each surface to determine the closest (visible) surface.
- In this method each surface is processed separately one pixel position at a time across the surface.
- The depth values for a pixel are compared and the closest (smallest z) surface determines the color to be displayed in the frame buffer.
- It is applied very efficiently on surfaces of polygon. Surfaces can be processed in any order. To override the closer polygons from the far ones, two buffers named frame buffer and depth buffer, are used.



- Depth buffer is used to store depth values for (x, y) position, as surfaces are processed ($0 \leq \text{depth} \leq 1$).
- The frame buffer is used to store the intensity value of color value at each position (x, y) .
- The z-coordinates are usually normalized to the range $[0, 1]$. The 0 value for z-coordinate indicates back clipping pane and 1 value for z-coordinates indicates front clipping pane.

Algorithm

Step-1 – Set the buffer values –

Depthbuffer $(x, y) = 0$

Framebuffer $(x, y) = \text{background color}$

Step-2 – Process each polygon (One at a time)

For each projected (x, y) pixel position of a polygon, calculate depth z.

If $Z > \text{depthbuffer} (x, y)$

Compute surface color,

set depthbuffer $(x, y) = z$,

framebuffer $(x, y) = \text{surfacecolor} (x, y)$

Advantages

1. It is easy to implement.
2. It reduces the speed problem if implemented in hardware.
3. It processes one object at a time.

Disadvantages

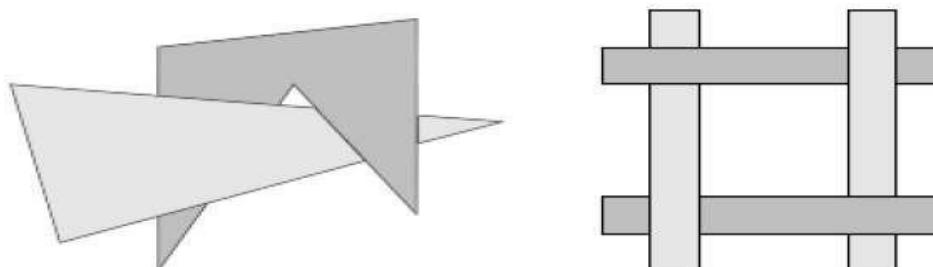
1. It requires large memory.
2. It is time consuming process.

Q. Explain Painters Algorithm (Depth Sorting Method)

Ans:

- Depth sorting method uses both image space and object-space operations. The depth-sorting method performs two basic functions –

1. First, the surfaces are sorted in order of decreasing depth.
 2. Second, the surfaces are scan-converted in order, starting with the surface of greatest depth.
- The scan conversion of the polygon surfaces is performed in image space.
 - This method for solving the hidden-surface problem is often referred to as the painter's algorithm. The following figure shows the effect of depth sorting –



- The algorithm begins by sorting by depth. For example, the initial $\| \text{depth} \|$ estimate of a polygon may be taken to be the closest z value of any vertex of the polygon.
- Let us take the polygon P at the end of the list. Consider all polygons Q whose z-extents overlap P's. Before drawing P, we make the following tests. If any of the following tests is positive, then we can assume P can be drawn before Q.
 - Do the x-extents not overlap?
 - Do the y-extents not overlap?
 - Is P entirely on the opposite side of Q's plane from the viewpoint?
 - Is Q entirely on the same side of P's plane as the viewpoint?
 - Do the projections of the polygons not overlap?
- If all the tests fail, then we split either P or Q using the plane of the other. The new cut polygons are inserting into the depth order and the process continues. Theoretically, this partitioning could generate $O(n^2)$ individual polygons, but in practice, the number of polygons is much smaller.

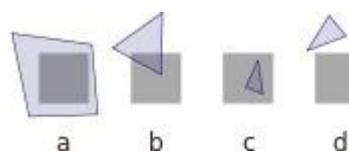
Q. Explain the Warnock's Algorithm.

Ans:

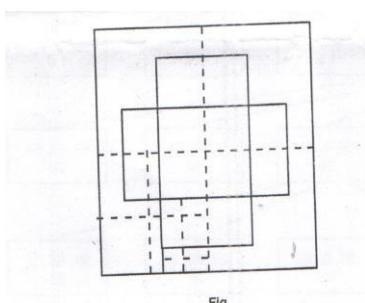
The Warnock algorithm is a hidden surface algorithm invented by John Warnock that is typically used in the field of computer graphics.^[1] It solves the problem of rendering a complicated image by recursive subdivision of a scene until areas are obtained that are trivial to compute. In other words, if the scene is simple enough to compute efficiently then it is rendered; otherwise it is divided into smaller parts which are likewise tested for simplicity.^[2]

This is a divide and conquer algorithm with run-time of $O(n \log n)$, where n is the number of polygons and p is the number of pixels in the viewport.

The inputs are a list of polygons and a viewport. The best case is that if the list of polygons is simple, then draw the polygons in the viewport. Simple is defined as one polygon (then the polygon or its part is drawn in appropriate part of a viewport) or a viewport that is one pixel in size (then that pixel gets a color of the polygon closest to the observer). The continuous step is to split the viewport into 4 equally sized quadrants and to recursively call the algorithm for each quadrant, with a polygon list modified such that it only contains polygons that are visible in that quadrant.



.Polygon visibility in a given viewport: a) polygon fills the viewport, b) polygon partially and c) completely visible, d) polygon invisible.



Q. Explain Rendering Process

Ans:

- Rendering is the process of producing realistic images or pictures.
- Producing realistic images involves both physics and psychology. Light, i.e., electromagnetic energy, reaches the eye after interacting with the physical environment. The human brain interprets this information for visualization of the object.
- Therefore, to produce realistic images we should know which surface of the object appears brighter or darker and which surface appears lighter.
- Many times we have to show the shadows of the objects appear in the picture, i.e., realistic displays of objects are obtained by generating perspective projections with hidden surface removed.

Illumination Models

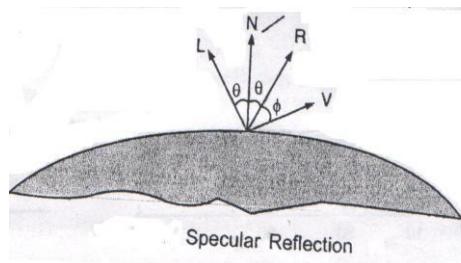
- Here we concentrate on the location and qualities of the light that falls on the objects and the way in which object interacts with it. A model for the interaction of light with a surface is called an illumination model.

Diffuse Illumination

- We will assume that our object is illuminated by light which does not come from any particular source but which comes from all directions. i.e., ambient light.
- When such illumination is uniform from all directions, the illumination is called diffuse illumination.
- Actually, diffuse illumination is a background light which is reflected from walls, floor and ceiling.
- We will assume that there is as much light going up as there is going down and that there is the same amount going right as there is going left.
- When we assume that going up, down right and left is of same amount then we can say that the reflection are constant over each surface of the object and they are independent of the viewing direction. Such a reflection is called diffuse reflection.

Q. Explain Specular reflection.

Ans:



- When we look at an illuminated shiny surface such as polished metal or a person's forehead, we see a highlight, or bright spot, at certain viewing directions. This phenomenon, called specular reflection, is the result of total or near total, reflection of the incident light in a Concentrated region around the specular reflection angle.
- Due to specular reflection the surface appears to be not in its original colour, but white, the colour of incident light.
- The fig. shows the specular reflection direction at a point on the Illuminated surface. The specular reflection angle equals the angle of the incident light, with the two angles measured when vector V and R coincide, i.e., $\phi = 0$.

Q. Explain Constant-Intensity Shading (Flat Shading)

Ans:

The fast and simplest method for shading polygon is constant shading, also known as faceted shading or flat shading. In this method, illumination model is applied only once for each polygon to determine single intensity value. The entire polygon is then displayed with the single intensity value.

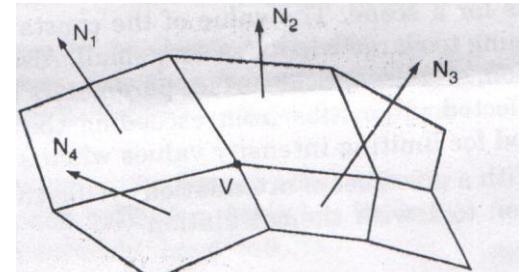
This method is valid for the following assumptions:

The light source is at infinity, so $N \cdot L$ is constant across the polygon face.

The viewer is at infinity, so $V \cdot R$ is constant over the surface.

The polygon represent the actual surface being modeled , and is not an approximation to a curved surface.

If either of the first two assumption are not true still we can use constant intensity shading approach.; however, we require some method to determine a single value for each of L and V Vectors



Q. Explain Phong Shading.

Ans:

A more accurate method for rendering a polygon surface is to interpolate normal vectors and then apply the illumination model to each surface point. This method, developed by Phong Bui Tuong, is called Phong shading or normal- vector interpolation shading. It interpolates the surface normal vector N , instead of the intensity.

By performing following steps, we can display polygon surface using Phong shading.

Determine the average unit normal vector at each polygon vertex.
Linearly interpolate the vertex normals over the surface of the polygon.
Apply an illumination model along each scan line to determine projected pixel intensities for the surface points

Advantages

1. It displays more realistic highlights on a surface.
2. It greatly reduces the Mach-band effect.
3. It gives more accurate result.

Disadvantages

1. It requires more calculation and greatly increases the cost of shading steeply.

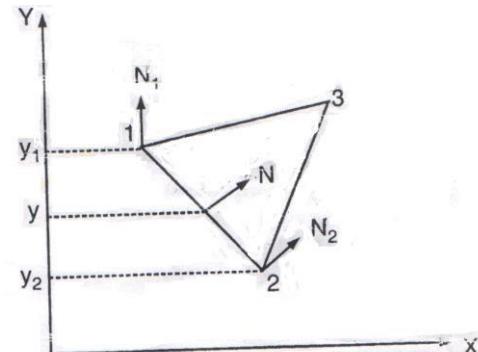


Fig. Calculation of interpolation of surface normals along a polygon edge

Q. Explain Shadow.

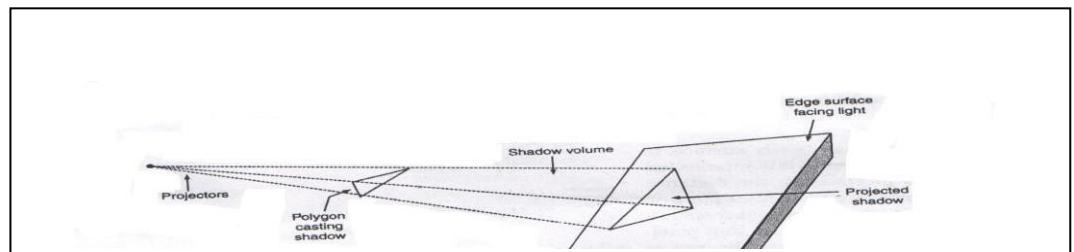
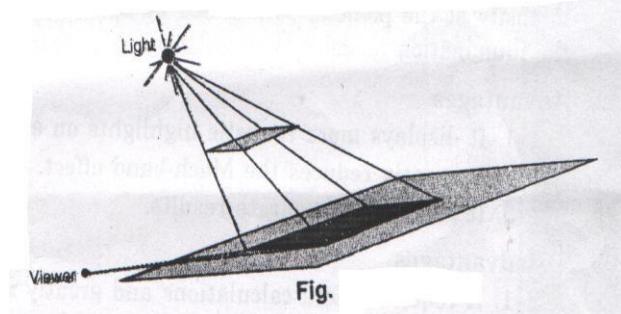
Ans:

A shadowed object is one where light is blocked from reaching i.e., the object which is hidden with respect to the light source. However, Shadows are seen only when observer's position is not concides with light source. The shadowed areas are not visible from the light source position but are visible from observer's position. Such areas can be identified by applying the hidden surface detection methods with view point assumed at the light source position.

A shadowed object is one which is hidden from the light source. It is possible to use hidden surface algorithms to locate the areas where light sources produce shadows. In order to achieve this we have to repeat the hidden-surface calculation using light source as viewpoint. This calculation divides the surfaces into shadowed and unshadowed groups.

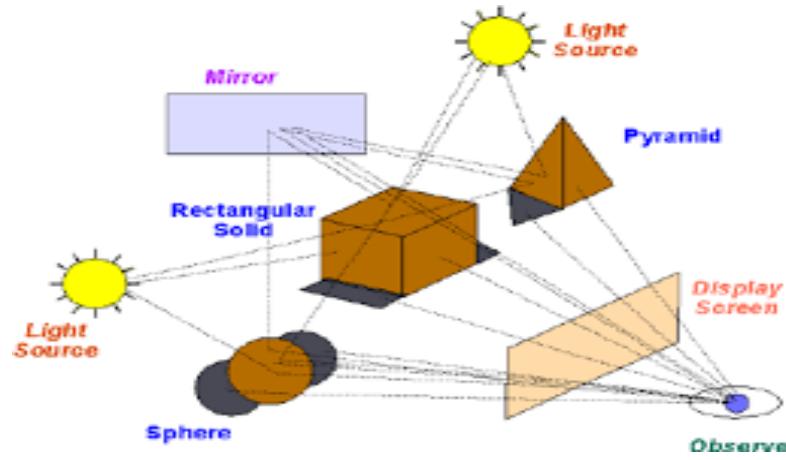
There are two types of shadows: shelf shadow and projected shadow. Shelf-shadows are created on some of the object planes when the object itself occludes light from illuminating those planes.

A projected shadow result on a surface when an intervening object blocks light from reaching the surface.

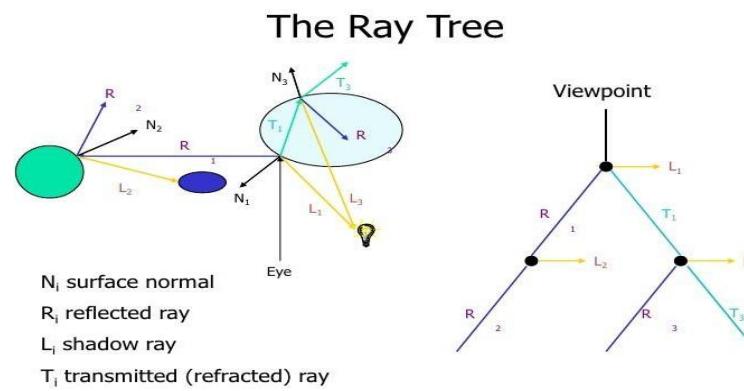


Q. Explain Basic Ray Tracing algorithm

Ans:



- We setup the coordinate system with pixel position designed in the xy plane and projection reference point lie on the z axis.
- From center of projection we then determine a ray path that passes through the center of each screen pixel position. In basic ray tracing algorithm, we consider one ray per pixel which equivalent to viewing scene through pinhole camera.
- For each pixel ray we test each surface is tested in the picture to determine if it is intersected by the ray.
- If surface is intersected, the distance from the pixel to the surface intersection point is calculated.
- The smallest calculated intersection distance identifies visible surface for that pixel.
- Once the visible surface identifies the ray is reflected off the visible surface along a specular path where the angle reflection equals angle incidence.
- If the surface is transparent, the ray is passed through the surface in the reflection direction.
- The ray reflected from the visible surface or passed through the transparent surface in refraction direction is called secondary ray.
- The ray after reflection or refraction strikes another visible surface.
- This process is repeated for each secondary ray.
- Objects are tested for intersection and the nearest surface along secondary ray path is used to recursively produce next generation of reflection and refraction paths.
- These paths are represented by ray tracing tree.
- As shown in figure, we use left branches in tree to represent reflection paths and right branches represent transmission path.
- Maximum depth of ray tracing trees can be set as a user option or it can be determined by the amount of storage available.
- A path n tree is then terminated if it reaches present maximum or if the ray strikes the light.



Q. Explain Types of Curves.

Ans:

A curve is an infinitely large set of points. Each point has two neighbors except endpoints. Curves can be broadly classified into three categories – **explicit, implicit, and parametric curves.**

Implicit Curves

Implicit curve representations define the set of points on a curve by employing a procedure that can test to see if a point is on the curve. Usually, an implicit curve is defined by an implicit function of the form –

$$f(x, y) = 0$$

It can represent multivalued curves (multiple y values for an x value). A common example is the circle, whose implicit representation is

$$x^2 + y^2 - R^2 = 0$$

Explicit Curves

A mathematical function $y = f(x)$ can be plotted as a curve. Such a function is the explicit representation of the curve. The explicit representation is not general, since it cannot represent vertical lines and is also single-valued. For each value of x, only a single value of y is normally computed by the function.

Parametric Curves

Curves having parametric form are called parametric curves. The explicit and implicit curve representations can be used only when the function is known. In practice the parametric curves are used. A two-dimensional parametric curve has the following form –

$$P(t) = f(t), g(t) \text{ or } P(t) = x(t), y(t)$$

The functions f and g become the (x, y) coordinates of any point on the curve, and the points are obtained when the parameter t is varied over a certain interval [a, b], normally [0, 1].

Bezier Curves

Bezier curve is discovered by the French engineer **Pierre Bézier**. These curves can be generated under the control of other points. Approximate tangents by using control points are used to generate curve. The Bezier curve can be represented mathematically as –

$$\sum k=0^n P_i B_{n i}(t)$$

Where P_i is the set of points and $B_{n i}(t)$ represents the Bernstein polynomials which are given by –

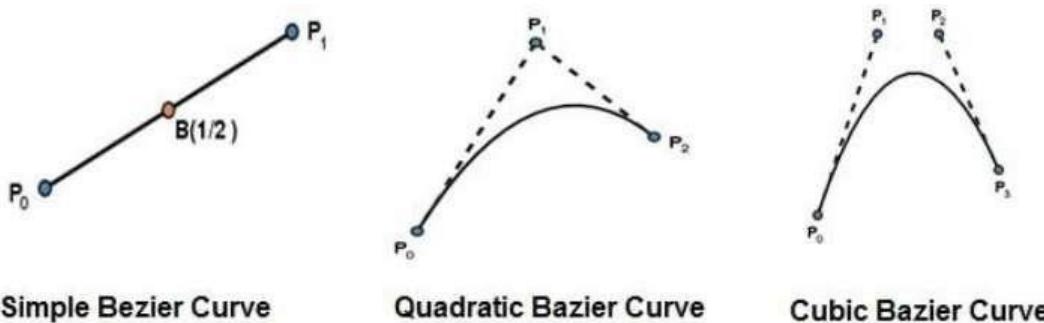
$$B_{n i}(t) = (n)_i (1-t)^{n-i} t^i$$

Where n is the polynomial degree, i is the index, and t is the variable.

The simplest Bézier curve is the straight line from the point P_0 to P_1 .

A quadratic Bezier curve is determined by three control points.

A cubic Bezier curve is determined by four control points.



Properties of Bezier Curves

Bezier curves have the following properties –

1. They generally follow the shape of the control polygon, which consists of the segments joining the control points.
2. They always pass through the first and last control points.
3. They are contained in the convex hull of their defining control points.

4. The degree of the polynomial defining the curve segment is one less than the number of defining polygon point. Therefore, for 4 control points, the degree of the polynomial is 3, i.e. cubic polynomial.
5. A Bezier curve generally follows the shape of the defining polygon.
6. The direction of the tangent vector at the end points is same as that of the vector determined by first and last segments.
7. The convex hull property for a Bezier curve ensures that the polynomial smoothly follows the control points.
8. No straight line intersects a Bezier curve more times than it intersects its control polygon.
9. They are invariant under an affine transformation.
10. Bezier curves exhibit global control means moving a control point alters the shape of the whole curve.
11. A given Bezier curve can be subdivided at a point $t=t_0$ into two Bezier segments which join together at the point corresponding to the parameter value $t=t_0$.

B-Spline Curves

The Bezier-curve produced by the Bernstein basis function has limited flexibility.

- First, the number of specified polygon vertices fixes the order of the resulting polynomial which defines the curve.
- The second limiting characteristic is that the value of the blending function is nonzero for all parameter values over the entire curve.

The B-spline basis contains the Bernstein basis as the special case. The B-spline basis is non-global.

A B-spline curve is defined as a linear combination of control points P_i and B-spline basis function $N_{i,k}(t)$ given by $C(t) = \sum_{i=0}^{n-k} P_i N_{i,k}(t)$, $t \in [t_{k-1}, t_n]$

Where,

- $\{P_i : i=0, 1, 2, \dots, n\}$ are the control points
- k is the order of the polynomial segments of the B-spline curve. Order k means that the curve is made up of piecewise polynomial segments of degree $k-1$,
- the $N_{i,k}(t)$ are the —normalized B-spline blending functions|. They are described by the order k and by a non-decreasing sequence of real numbers normally called the —knot sequence|.

$t_i : i=0, \dots, n+K$

The $N_{i,k}$ functions are described as follows –

$$N_{i,1}(t) = \begin{cases} 1, & \text{if } t \in [t_i, t_{i+1}) \\ 0, & \text{Otherwise} \end{cases}$$

and if $k > 1$,

$$N_{i,k}(t) = \frac{t - t_i}{t_{i+k-1} - t_i} N_{i,k-1}(t) + \frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(t)$$

and

$$t \in [t_{k-1}, t_n]$$

Properties of B-spline Curve

B-spline curves have the following properties –

1. The sum of the B-spline basis functions for any parameter value is 1.
2. Each basis function is positive or zero for all parameter values.
3. Each basis function has precisely one maximum value, except for $k=1$.
4. The maximum order of the curve is equal to the number of vertices of defining polygon.
5. The degree of B-spline polynomial is independent on the number of vertices of defining polygon.
6. B-spline allows the local control over the curve surface because each vertex affects the shape of a curve only over a range of parameter values where its associated basis function is nonzero.
7. The curve exhibits the variation diminishing property.
8. The curve generally follows the shape of defining polygon.
9. Any affine transformation can be applied to the curve by applying it to the vertices of defining polygon.
10. The curve line within the convex hull of its defining polygon.

Q. Explain Polygon Meshes.

Ans:

3D surfaces and solids can be approximated by a set of polygonal and line elements. Such surfaces are called **polygons meshes**. In polygon mesh, each edge is shared by at most two polygons. The set of polygons or faces, together form the —skin|| of the object.

This method can be used to represent a broad class of solids/surfaces in graphics. A polygonal mesh can be rendered using hidden surface removal algorithms. The polygon mesh can be represented by three ways –

- Explicit representation
- Pointers to a vertex list
- Pointers to an edge list

Advantages

- It can be used to model almost any object.
- They are easy to represent as a collection of vertices.

- They are easy to transform.
- They are easy to draw on computer screen.

Disadvantages

- Curved surfaces can only be approximately described.
- It is difficult to simulate some type of objects like hair or liquid.

Q. Explain 3D Transformation.

Ans:

Transformation matrix is a basic tool for transformation. A matrix with $n \times m$ dimensions is multiplied with the coordinate of objects. Usually 3×3 or 4×4 matrices are used for transformation. For example, consider the following matrix for various operation.

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix}$$

Translation Matrix

$$S = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Scaling Matrix

$$Sh = \begin{bmatrix} 1 & sh_x^y & sh_x^z & 0 \\ sh_y^x & 1 & sh_y^z & 0 \\ sh_z^x & sh_z^y & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Shear Matrix

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation Matrix

Transformation

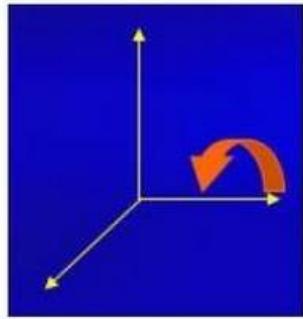
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix}$
Translation Matrix

Rotation

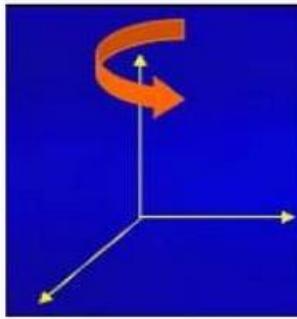
3D rotation is not same as 2D rotation. In 3D rotation, we have to specify the angle of rotation along with the axis of rotation. We can perform 3D rotation about X, Y, and Z axes. They are represented in the matrix form as below –

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

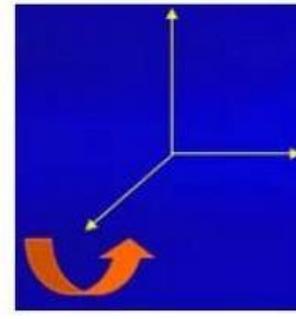
The following figure explains the rotation about various axes –



Rotation about x-axis



Rotation about y-axis



Rotation about z-axis

Scaling

You can change the size of an object using scaling transformation. In the scaling process, you either expand or compress the dimensions of the object. Scaling can be achieved by multiplying the original coordinates of the object with the scaling factor to get the desired result. The following figure shows the effect of 3D scaling .

In 3D scaling operation, three coordinates are used. Let us assume that the original coordinates are (X, Y, Z), scaling factors are (SX,SY,Sz) respectively, and the produced coordinates are (X', Y', Z'). This can be mathematically represented as shown below –

$$S = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P' = P \cdot S$$

Shear

A transformation that slants the shape of an object is called the **shear transformation**. Like in 2D shear, we can shear an object along the X-axis, Y-axis, or Z-axis in 3D.

$$Sh = \begin{bmatrix} 1 & sh_x^y & sh_x^z & 0 \\ sh_y^x & 1 & sh_y^z & 0 \\ sh_z^x & sh_z^y & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P' = P \cdot Sh$$

Q. Explain Polygon Meshes.

Ans:

3D surfaces and solids can be approximated by a set of polygonal and line elements. Such surfaces are called **polygonal meshes**. In polygon mesh, each edge is shared by at most two polygons. The set of polygons or faces, together form the –skin|| of the object.

This method can be used to represent a broad class of solids/surfaces in graphics. A polygonal mesh can be rendered using hidden surface removal algorithms. The polygon mesh can be represented by three ways –

- Explicit representation
- Pointers to a vertex list
- Pointers to an edge list

Advantages

- It can be used to model almost any object.
- They are easy to represent as a collection of vertices.
- They are easy to transform.
- They are easy to draw on computer screen.

Disadvantages

- Curved surfaces can only be approximately described.
- It is difficult to simulate some type of objects like hair or liquid.