CS-21302
Computer Graphics
Handouts

Computer Graphics

Introduction:

This handout provides supporting material of computer graphics to accompany with the video lectures. The content is mainly delivered in the lectures and this handout will follow the same order as that of used in video lectures.

Computer Graphics: The study of using computers to create and manipulate images is called Computer Graphics. Storage of geometric objects and their images also fall in category of Computer Graphics.

Graphics Areas

Modeling

The area of Computer graphics that deals with the mathematical specification of shape and appearance properties in a way that can be stored on the computer is called Modeling.

Rendering

The creation of shaded images from 3D computer models is called Rendering.

Illumination

Modeling of interaction of light with matter is called Illumination.

Animation

The technique to create an illusion of motion through sequences of images is referred to as Animation.

Applications

Computer Graphics are utilized in almost every wake of life. Graphics are used in for

- Entertainment
    - Video Games
    - Cartoons
    - Animated Films
- Industry
    - CAD/CAM: These fields use computer technology to design parts and products on the computer and then, using these virtual designs, to guide the manufacturing process, e. g. AutoCAD from AUTODesk Microsoft Visio and Adobe's Photoshop etc.
- Simulations
    - Can be taken as accurate video games e.g. Flight Simulator, Power Plant Simulator, Firefighter Simulator
- Medicine

    - Medical Imaging
    - Meaningful images of patients scanned data
    - MRI
    - CT Scan
    - Guided Surgery
- Scientific Visualization
- Education and Training
- Everyday Uses

Visualization

Representing data with graphical illustrations is called *visualization.* Visualization can be categorized as

- Producing graphical representations for scientific, engineering, and medical data sets and processes is generally referred to as *scientific visualization*
- The *business visualization is used* in connection with data sets related to commerce, industry, and other nonscientific areas.

Image Processing

Image processing applies techniques to modify or interpret existing pictures, enhancing images and videos

Machine vision

Machine perception of the visual information (robotics)

Graphical User Interfaces (GUI)

Graphical User Interfaces display menus and icons for fast selection of any processing required.

The user does not have to memorize commands or capabilities of computer software. All options provided by the software can be explored just by navigating using mouse clicks.

A windows manager allows user to display multiple windows. Each window can process different data (weather graphical or text). A window can be made active just by clicking mouse on it.

Output Devices

Cathode Ray Tube (CRT)

Filament (acts as heating element, electrons emit from it in the form of beam). Electrons move towards positive plate (Anode) focusing cylinder (because of Electric field). Vertical and horizontal deflection plates have magnetic field in between them and control the position of the electron beam. The beam strikes phosphor coating on front of tube producing illumination. Stronger the beam, brighter is the screen. Longer the beam stays on a point brighter is the screen.

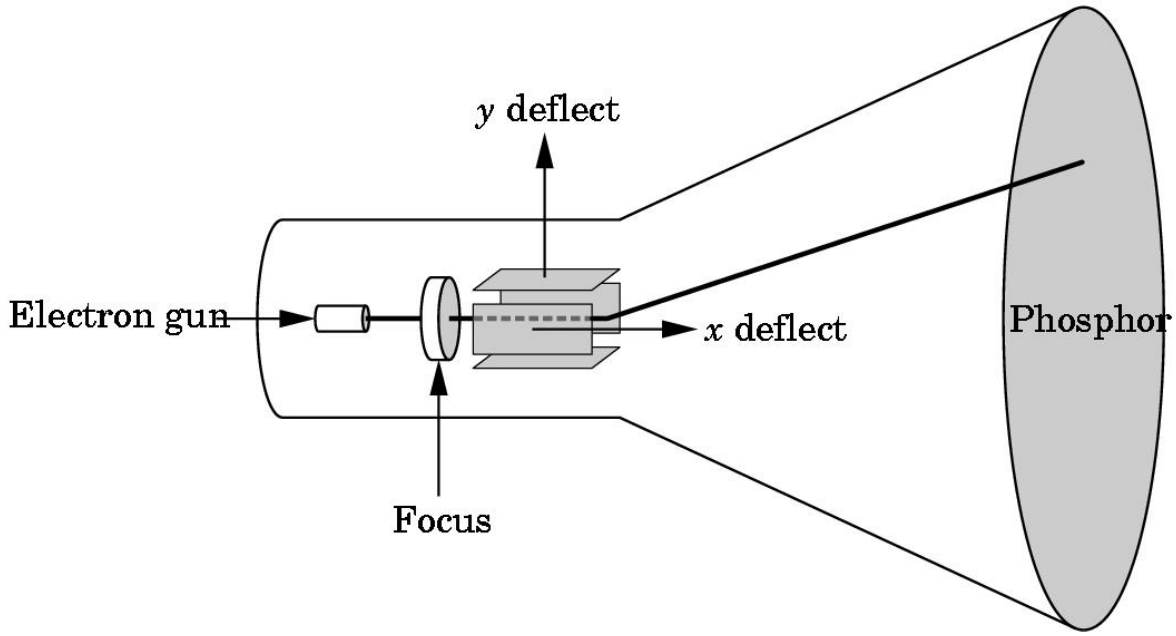


Figure 1: Cathode Ray Tube

Characteristics of CRT

It's massive evacuated glass tube. Capabilities of CRT are measured by

- Size of tube
- Brightness of the phosphors vs. darkness of the tube
- Speed of electron gun
- Diameter of the beam
- Pixels
- Disadvantages are
- Size of the tube
- High power consumption
- Flicker
- Costly to refresh

Disadvantages of CRT are

- Huge size of the tube
- High power consumption
- Flicker
- Costly to refresh

Florescence is Light emitted while the phosphor is being struck by electrons

Phosphorescence is Light emitted once the electron beam is removed

Persistence: The time from the removal of the excitation to the moment when phosphorescence has decayed to 10% of the initial light output

Interlacing:

A frame is divided into odd (only odd numbered lines are shown at an instant) and even (only even numbered lines are shown at next instant) fields as shown in the Figure 2

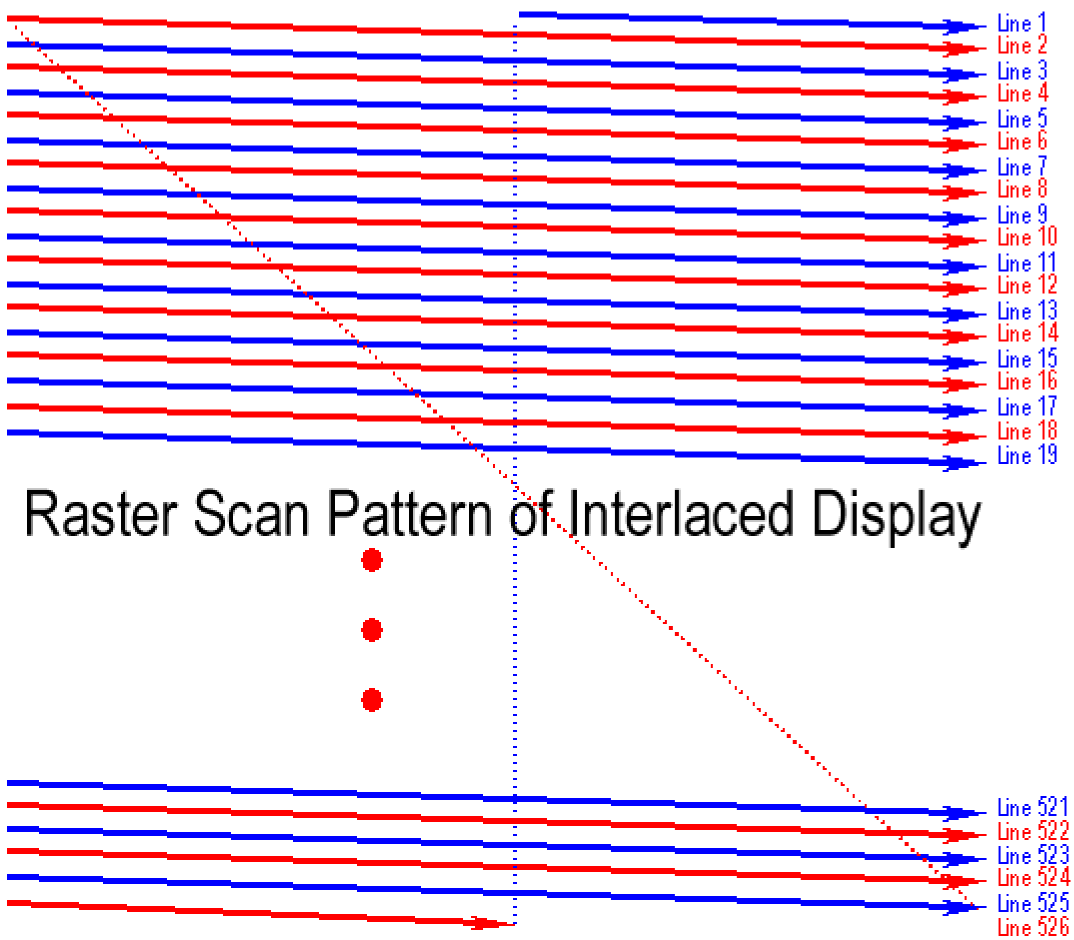


Figure 2: Raster Scan

Video Formats

Following video formats are used

- NTSC - 525x480, 30f/s, interlaced
- PAL - 625x480, 25f/s, interlaced
- VGA - 640x480, 60f/s, non-interlaced
- SVGA – 800x600, 60f/s non-interlaced

- RGB - 3 independent video signals and synchronization signal, vary in resolution and refresh rate
- Time-multiplexed color - R,G,B one after another on a single signal, vary in resolution and refresh rate

Raster Displays

A rectangular array of points or dots is called Raster.

A Pixel is picture element of raster.

A row of pixels is known as Scan Line.

Picture elements stored in memory are called Frame Buffer.

Raster displays

- Use sequential access
- Raster Displays (early 70s)
- like television, scan all pixels in regular pattern
- use frame buffer (video RAM) to eliminate sync problems

Frame Buffer

Raster images require frame buffers. Frame buffer - A block of memory, dedicated to graphics output, that holds the contents of what will be displayed. If we want a frame buffer of 640 pixels by 480 pixels, we should allocate a frame buffer = 640*480 bits.

Movie Standards

Films play at 24 fps by U.S. film projectors. Projectors have a shutter to block light during frame advance. To reduce flicker, shutter opens twice for each frame – resulting in 48 fps flashing which is perceptually acceptable. European film projectors play film at 25 fps. American films are played 'as is' in Europe, resulting in everything moving 4% faster. Faster movements and increased audio pitch are considered perceptually acceptable.

Movies can be converted to be watched at TV as under.

A 24 film fps must be converted to NTSC U.S. television interlaced 29.97 fps 768x494 or PAL Europe television 25 fps 752x582. Use 3:2 Pull – down. First frame of movie is broken into first three fields (odd, even, odd), Next frame of movie is broken into next two fields (even, odd), Next frame of movie is broken into next three fields (even, odd, even) and so on.

Liquid Crystal Display

Liquid crystal displays use small flat chips which change their transparency properties when a voltage is applied. LCD elements are arranged in an n x m array call the LCD matrix. Level of voltage controls gray levels. LCDs elements do not emit light, use backlights behind the LCD matrix. Color is obtained by placing filters in front of each LCD element. Image quality depends on viewing angle.

Advantages of LCD

LCD's are Flat, Lightweight and consume low power.

Resolution

The Resolution is defined as the number of dot per inch or centimeter that can be plotted horizontally & vertically. Higher the resolution, smaller is the spot size. Resolution is related with quality of the graphics system. Higher the resolution better is the graphics system. High quality resolution is 1280x1024, for example. The intensity distribution of spots on the screen has Gaussian distribution. Adjacent points will appear distinct as long as their separation is greater than the diameter at which each spot has intensity of about 60% of that at the center of the spot.

Aspect Ratio

The Aspect Ratio is the ratio between number of scan lines in a rater and the number of pixels in a scan line necessary to produce equal length lines in both directions on the screen.

For example 1280 x 768 has an aspect ratio of 5:3.

LED Display

LED, actually, stands for Light Emitting Diode. LED displays use the same technology for display but it uses LED's for back light while LCD displays generally use Cold Cathode Fluorescent technology

Plasma Panel

Plasma display panels are similar in principle to fluorescent light tubes. Small gas-filled capsules are excited by electric field, emits UV light, and UV excites phosphor, phosphor relaxes, emits some other color.

Plasma panels are fairly bright and offer large viewing angle and are good for large format display. Plasma panels, however, are expansive, have large pixel size (~1mm as compared to ~0.2mm). Phosphor depletes with time and are less bright than CRT

Vector Display

The electron beam illuminates only the parts of the screen where a picture is to be drawn. It works like a plotter i.e. draws a one picture line at a time and is used in line drawing and wireframe displays. Picture definition is stored as a set of line-drawing commands stored in a refresh display file. Number of lines derive Refresh rate and refresh cycle which is between 30 and 60 per second. Vector display can draw 100,000 short lines at this refresh rate.

Resolution of vector displays is higher than raster and line drawings are smooth while these displays are not suitable for realistic shaded scenes.

Hardcopy Devices

Following hard copy devices are used with graphics systems

- Inkjet Printer
- Laser Printer
- Film Recorder
- Electrostatic Printer and
- Pen Plotter

Input Devices

Locator Devices: used to indicate a position and/or orientation and to select a displayed entity e.g. Tablet, Mouse, Trackball, Joystick, Touch Panel, Light Pen

Keyboard devices: to input a character string e.g. Alphanumeric keyboard (coded - get single ASCII character, un-encoded - get state of all keys - more flexible)

Valuator Devices: to input a single value in the space of real numbers Rotary dials (Bounded or Unbounded), such as Linear sliders.

Choice Devices: to select from a set of possible actions or choices

Function keys

Human Visual System (HVS)

Rods and cones are energized by electromagnetic energy in the range 350-780 nm. Size (number) of rods and cones determines the resolution of HVS – our visual acuity. The sensors in the human eye do not react uniformly to the light energy at different wavelengths. HVS responds differently for single frequency light – red/green/blue.

Ray Tracing

Building an imaging model by following light from a source is known as Ray Tracing. A Ray is a semi – infinite line that emerges from s source and continues to infinity in one direction. Parts of ray that contribute in making image are computed by the contribution from all arrays whether they are incident directly or reflected from other objects.

Surfaces of the objects may be diffusing, reflecting or refracting. For each pixel intensity must be computed, contributions of all rays must be taken into account.

Many algorithms exist to make this not only feasible, but remarkably efficient. Tracing one ray is a complex problem and requires serious work to make it run at an acceptable speed. Of course, the big problem is the fact that one needs to trace lots of rays to generate a high quality image.

Lighting

Once we have the key intersection information (position, normal, color, texture coordinates, etc.) we can apply any lighting model we want. This can include procedural shaders, lighting computations, texture lookups, texture combining, bump mapping, and more. Many of the most interesting forms of lighting involve spawning off additional rays and tracing them recursively. The result of the lighting equation is a color, which is used to color the pixel

Ray generation, which computes the origin and direction of each pixel's viewing ray based on the camera geometry.

A ray when intercepted by a surface splits into two rays, Absorbed or Reflected.

Shading, which computes the pixel color based on the results of ray intersection.

Pixel Formats

There are many pixel formats following pixel formats are more common.

- 1-bit gray scale—text and other images where intermediate grays are not desired (high resolution required)
- 8-bit RGB fixed-range color (24 bits total per pixel)—web and email applications, consumer photographs
- 8- or 10-bit fixed-range RGB (24–30 bits/pixel)—digital interfaces to computer displays;
- 12- to 14-bit fixed-range RGB (36–42 bits/pixel)—raw camera images for professional photography
- 16-bit fixed-range RGB (48 bits/pixel)—professional photography and printing, intermediate format for image processing of fixed-range images;
- 16-bit fixed-range gray scale (16 bits/pixel)—radiology and medical imaging;
- 16-bit "half-precision" floating-point RGB—HDR images; intermediate format for real-time rendering;
- 32-bit floating-point RGB—general-purpose intermediate format for software rendering and processing of HDR images.

Image Storage

RGB images result in almost 3MB for 1 million pixel image. To reduce storage space, various compression techniques are employed and used in image compression. Generally compression is categorized as

- Lossless Compression (image loses no information)

- Lossy Compression (some quality may be compromised)

Storage Formats

Few common formats are as under

- jpeg. This lossy format compresses image blocks based on thresholds in the human visual system. This format works well for natural images.
- tiff. This format is most commonly used to hold binary images or losslessly compressed 8- or 16-bit RGB although many other options exist.
- ppm. (Portable Pixel Map)This very simple lossless, uncompressed format is most often used for 8-bit RGB images although many options exist.
- png. (Portable Network Graphics) This is a set of lossless formats with a good set of open source management tools.

Views:

Perspectives: The process of creating 2D images of 3D objects/scenes. In linear perspective, 3D objects are projected onto an image plane in such a way that straight lines in the scene become straight lines in the image.

Parallel Projection: 3D points are mapped to 2D by moving them along a *projection direction* until they hit the image plane

Projections

When projection lines are parallel and perpendicular to the image plane, the resulting views are called orthographic. The image plane is perpendicular to the view direction, the projection is called *orthographic* otherwise it is called oblique

Creating View

In three-point perspective, an artist picks "vanishing points" where parallel lines meet. Parallel horizontal lines will meet at a point on the horizon. Every set of parallel lines has its own vanishing points. These rules are followed automatically if we implement perspective based on the correct geometric principles.

Computing Viewing Rays

We start by 'shooting' rays from the camera out into the scene. We can render the pixels in any order we choose (even in random order!), but we will keep it simple and go from top to bottom, and left to right. We loop over all of the pixels and generate an initial *primary ray* (also called a *camera ray* or *eye ray*). The ray origin is simply the camera's position in world space. The direction is computed by first finding the 4 corners of a virtual image in world space, then interpolating to the correct spot, and finally computing a normalized direction from the camera position to the virtual pixel.

Shadow Rays

Shadow rays behave slightly differently from primary (and secondary) rays. Normal rays (primary & secondary) need to know the first surface hit and then compute the color reflected off of the surface. Shadow rays, however, simply need to know if something is hit or not. In other words, we don't need to compute any additional shading for the ray and we don't need to find the closest surface hit. This makes them a little faster than normal rays.

Viewing Problems

The problems related to viewing are

- Where a viewer is located (Location)
- Viewing Plane
- The visible portion of the scene
- e., what can be seen (clipping)
- Maintaining relation between objects
- Parallel lines
- Angles
- Distances
- Relation to the viewer
- Scenes and Objects
- Viewing a scene or object

Some viewing techniques perform better in viewing objects than a whole scene.

External Viewing: Viewing an object from outside e. g. a building

Internal Viewing: Viewing from inside. e.g. internal of a building specially in games.

Projections

Project N-Dimensions Coordinates onto <N Dimensions Coordinates. When projection lines are parallel and perpendicular to the image plane, the resulting views are called orthographic. The image plane is perpendicular to the view direction, the projection is called *orthographic* otherwise it is called oblique

Perspective

The process of creating 2D images of 3D objects/scenes. In linear perspective, 3D objects are projected onto an image plane in such a way that straight lines in the scene become straight lines in the image.

Parallel Projection: 3D points are mapped to 2D by moving them along a *projection direction* until they hit the image plane.

Views are classified in

- Parallel
- Orthographic: The projectors are orthogonal to the projection surfaces
- Top (Plan): Projection planes can move relative to the object
- Axonometric: Projection planes can move relative to the object. Axonometric projections are classified by how many angles of a corner of a projected cube are the same.
- Trimetric (none of the angles is same)
- Dimetric (two angles are same)
- Isometric (all three angles are same)
- Oblique Projections: Relationship between projectors and projection plane is arbitrary
- Perspective Projections: Perspective views are characterized by **diminution** of size. When objects are moved farther from the viewer, their images become smaller
- Three Point Perspective: None of the principal face is parallel to projection plane. No. of vanishing points is three in a cube
- Two Point Perspective: One of the principal directions is parallel to projection plane. No. of vanishing points is two in a cube
- One Point Perspective: One of the principal faces is parallel to projection plane. No. of vanishing points is one in a cube

Scalars, Points and Vectors

Geometry is a subject that relates objects in n – dimensional space (In computer graphics we deal with 3 dimensional spaces). Scalars, vectors and points form minimum set of primitives and are used to build sophisticated objects.

A **point** is a location in space that neither has size nor shape. Real numbers (magnitudes) such as distance between two points are scalars.

Scalars: Scalars are members of sets which can be combined by addition and multiplication and obey associativity, commutivity and inverses axioms. Scalars don't possess any geometric properties e.g. Real and Complex numbers

Vectors: A quantity defined by magnitude and direction such as Velocity, Force etc. For computer graphics, a directed line segment (can be used for any vector) is most significant example.

Properties of Vectors

Vectors possess inverses that equal in magnitude but opposite in direction.

Vectors have no location

Scalar Multiplication: A vector can be multiplied by a scalar (magnitude changes only not the direction)

Zero vector is also defined with zero magnitude and undefined direction

Head to Tail Rule is used to add vectors

A **Euclidean space** is an extension of a vector space that adds a measure of size or distance and allows us to define such things as the length of a line segment.

An **affine space** is an extension of the vector space that includes an additional type of object: the point. Operations between vectors and points are allowed

Point – Vector Operations

Point – point subtraction yields

$v = P - Q$

Point – Vector addition yields

$P = Q + v$

Lines

Consider all points of the form

$P(\alpha) = P_0 + \alpha \mathbf{d}$

where P_0 is an arbitrary point d is an arbitrary vector and α is a scalar (that can vary over a range). This relation can be interpreted as the set of all points that pass through P_0 in the direction of the vector d.

Dot product:

The dot product of two vectors is a scalar.

Let v_1 and v_2 be two vectors

$v_1 . v_2 = v_1 v_2 Cos\theta$

Dot Products (Properties and uses)

Used to compute length (magnitude) of the vector

Normalization (finding unit vector)

Computing angle between two vector

Checking for orthogonality

Finding projection of a vector along another vector

Dot product is commutative and distributive

Cross Product

The product of two vectors (a and b) is another vector which is orthogonal to both the vectors (a and b). The new vector is called Cross Product or Vector Product of the two vectors. Right hand rule determines the direction of the product.

Transformations (2D)

Scaling

Scaling is achieved by multiplying each graphic component by a scalar. Scaling will be uniform if this scalar is same for all components. Different components are scaled differently, scaling in uniform. E. g

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

Rotation: Let a point (x, y) rotates through an angle θ and its new location is (x', y') as shown in the figure 3. The coordinates of new point can be calculated using following transformation matrix. (For proof watch the relevant lecture)

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

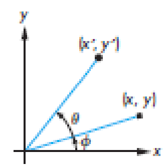


Figure 3: Rotation of a point in 2D

Translation

Shear

$$\text{shear-x}(s) = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}, \quad \text{shear-y}(s) = \begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix}.$$

Reflection

Homogenous Coordinate System

A 3rd Coordinate is added to every 2D point

- (x, y, t) represents (x/t, y/t)
- (x, y, 0) represents infinity

- (0, 0, 0) is not allowed

So in 2D coordinate system, 3 X 3 matrices are used

Transformations (3D): Transformations in 3D are similar to 2D and are as under

$$\text{scale}(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}.$$

$$\text{rotate-z}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{rotate-x}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

$$\text{rotate-y}(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}$$

$$\text{shear-x}(d_y, d_z) = \begin{bmatrix} 1 & d_y & d_z \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

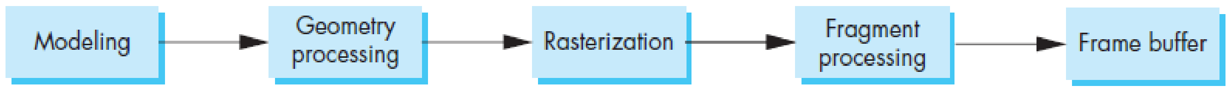
Rigid Body Transforms

The transforms in which angles and lengths are preserved are called rigid body transforms. The body or object is not distorted after the application of transformation. Rotation and Translation are examples

Affine Transformations

Parallelism of lines is preserved but angles between the lines are not preserved in affine transformations.

Implementation: Implementation can be explained by the following block diagram



Modeling: The usual results of the modeling process are sets of vertices that specify a group of geometric objects supported by the rest of the system.

Geometry Processing

Geometry processing means to determine which geometric objects can appear on the display and to assign shades or colors to the vertices of these objects.

Shading

Hidden surface removal and visible surface determination are required

Rasterization

Calculation of pixel values based upon the previous steps i. e. Projection, Primitive assembly, Clipping and Shading. The rasterizer starts with vertices in normalized device coordinates but outputs fragments whose locations are in units of the display—**window coordinates**.

Fragment Processing

Each fragment is assigned a color by the rasterizer and this color is placed in the frame buffer at the locations corresponding to the fragment's location. Some of the possibilities are merging with the results of the geometric pipeline at the rasterization stage, for example shaded and texture-mapped polygon is processed. Hidden-surface removal process is typically carried out on a fragment-by-fragment basis. Fragment colors may have to be blended with the colors of pixels already in the color buffer due to some translucent object

Clipping

Identification of the portions of geometric primitives by analytical calculations within the view windows in known as clipping. Not to clip means rasterize outside framebuffer and waste time to convert pixels outside the view window.

Line Clipping

A clipper decides which primitive, or part of primitives can possibly be displayed and be passed on to rasterizer. Primitives that fit within the specified view volume pass through the clipper, or are accepted.

Primitives that cannot appear on the display are eliminated, or rejected or culled. Primitives that are partially within the view volume must be clipped such that any part lying outside the volume is removed

Clipping Algorithms

- Test both end points of line
- Both ends on correct side of the view window (same) edge
- Trivial Accept

If both endpoints lie inside all the edges of view window, accept the line "trivially". If both endpoints lie outside the same edge of view window, reject the line "trivially". Otherwise, split the line in two segments and accept and reject each segment trivially.

Cohen – Sutherland Line Clipping

The view window is divided into regions as shown in the figure 4.

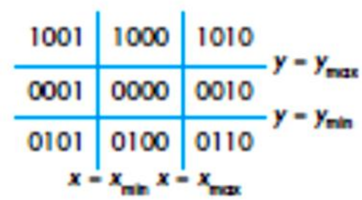


Figure 4: Space break up

Most significant bit (MSB) called bit 1 indicates y-value of points are above y_{max}

Bit 2 indicates y-value of points are below y_{min}

Bit 3 indicates x-value of points are to the right of X_{max}

Bit 4 indicates x-value of points are to the left of X_{min}

Consider a line segment whose outcodes are given by $o1$ = *outcode(x*1, *y*1*)* and$o2$ = *outcode(x*2, *y*2*)*. We can now reason on the basis of these outcodes. There are four cases

1. ($o1 = o2 = 0$). Both endpoints are inside the clipping window, as is true for segment *AB* in Figure. The entire line segment is inside, and the segment can be sent on to be rasterized.
2. ($o1 \neq 0$, $o2 = 0$; or vice versa). One endpoint is inside the clipping window; one is outside (see segment *CD* in Figure). The line segment must be shortened. The nonzero outcode indicates which edge or edges of the window are crossed by the segment. One or two intersections must be computed. Note that after one intersection is computed, we can compute the outcode of the point of intersection to determine whether another intersection calculation is required.

3. ($o1$ & $o2 \neq 0$). By taking the bitwise AND of the outcodes, we determine whether or not the two endpoints lie on the same outside side of the window. If so, the line segment can be discarded (see segment *EF* in Figure).
4. ($o1$ & $o2 = 0$). Both endpoints are outside, but they are on the outside of different edges of the window. As we can see from segments *GH* and *IJ* in Figure, we cannot tell from just the outcodes whether the segment can be discarded or must be shortened. The best we can do is to intersect with one of the sides of the window and to check the outcode of the resulting point.

Liang-Barsky Clipping

1. A line parallel to a clipping window edge has pi = 0 for that boundary.
2. If for that i, qi < 0, the line is completely outside and can be eliminated.
3. When pi < 0 the line proceeds outside to inside the clip window and when pi > 0, the line proceeds inside to outside.
4. For nonzero pi, t = qi/pi gives the intersection point.
5. For each line, calculate t1 and t2. For t1, look at boundaries for which pi < 0 (outside view window). Take t1 to be the largest among (0, qi/pi). For t2, look at boundaries for which pi > 0 (inside view window). Take t2 to be the minimum of (1, qi/pi). If t1 > t2, the line is outside and therefore rejected.

Polygon Clipping

Polygon clipping is difficult task. Polygon clipping may result in

1. Original Polygon
2. New polygon
3. Nothing

Sutherland-Hodgman Clipping

1. Consider each edge of the view window individually
2. Clip the polygon against the view window edge's equation
3. After doing all edges, the polygon is fully clipped
4. Light, Surface and Imaging
5. Illumination has strong impact on appearance of the surface

Lighting and Shading

Surfaces: Surfaces play important role in lighting and shading. Surfacetypes according to their reflective properties are as under.

Specular Surfaces: Specular surfaces appear shiny because most of the light that is reflected or scattered is in a narrow range of angles close to the angle of reflection. Mirrors are perfectly specular surfaces

Diffusive Surfaces: Diffuse surfaces are characterized by reflected light being scattered in all directions. Walls painted with matte or flat paint are diffuse reflectors. Perfectly diffuse surfaces scatter light equally in all directions. Flat perfectly diffuse surface appears the same to all viewers.

Translucent Surfaces: Translucent surfaces allow some light to penetrate the surface and to emerge from another location on the object. This process of refraction characterizes glass and water. Some incident light may also be reflected at the surface.

Shadows

Shadows created by finite-size light source. Shadows are classified as

1. Umbra – full shadow
2. Penumbra – partial shadow

Light Sources: Light sources are categorized as

1. Ambient Light (uniform lighting)
2. Point Source (emits light equally in all directions)
3. Spot Lights (Restrict light from ideal point source)

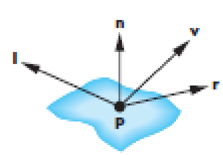
Phong Reflection Model

A simple model supports three models of light – matter interactions i.e.

1. Diffuse
2. Specular
3. Ambient

and uses four vectors

1. normal
2. to source
3. to viewer
4. perfect reflector



Ideal Reflector

Local orientation of the surface determines the normal vector. Law of reflection is obeyed. The three vectors must lie in same plane

Lambertian Surface

Perfectly diffuse reflector. Light scattered equally in all directions. Amount of light reflected is proportional to the vertical component of incoming light reflected light ~cos $\theta_{i.}$

cos θ_i = **l** · **n** if vectors normalized

There are also three coefficients, k_r, k_b, k_g that show how much of each color component is reflected

Specular Surfaces

Most surfaces are neither ideal diffusers nor perfectly specular (ideal reflectors). Specular highlights appear on smooth surfaces due to incoming light being reflected in directions close to the direction of a perfect reflection.

Specular Reflections Model

According to Phong, Reflected intensity $\mathbf{I}_r$ goes as absorption coeff. k_s and projection of incoming intensity along viewer (α is shinness coeff)

$\mathbf{I}_r \sim k_s \mathbf{I} \cos^{\alpha}\phi$

Values of α vary between 100 and 200 for metals. Values vary between 5 and 10 for surfaces that look like plastics.

Ambient Light

Ambient light is the result of multiple interactions between (large) light sources and the objects in the environment. Amount and color depend on both the color of the light(s) and the material properties of the object. Add k_a I_a (reflection coef * intensity of ambient light) to diffuse and specular terms

Distance Terms

The light from a point source that reaches a surface is inversely proportional to the square of the distance between them. We can add a factor of the form $1/(ad + bd + cd^2)$ to the diffuse and specular terms. The constant and linear terms soften the effect of the point source.

Light Sources

In the Phong Model, we add the results from each light source. Each light source has separate diffuse, specular, and ambient terms to allow for maximum flexibility even though this form does not have a physical justification. Separate red, green and blue components are included. Hence, 9 coefficients for each point source contribute.

I_{dr}, I_{dg}, I_{db}, I_{sr}, I_{sg}, I_{sb}, I_{ar}, I_{ag}, I_{ab}

Material Properties

Material properties match light source properties. Nine absorbtion coefficients also contribute.

k_{dr}, k_{dg}, k_{db}, k_{sr}, k_{sg}, k_{sb}, k_{ar}, k_{ag}, k_{ab}

For each light source and each color component, the Phong model can be written (without the distance terms) as

$$I = k_d I_d \; \mathbf{l} \cdot \mathbf{n} \; + k_s I_s (\mathbf{v} \cdot \mathbf{r})^{\alpha} + k_a I_a$$

For each color component we add contributions from all sources.

Graphical User Interfaces

A program may be judged by UI rather than its functionality. A poor interface may cause user to make errors. A poor UI may be a hurdle in making people use. Most of the system whether business or engineering are available with Graphical User Interfaces (GUI) thus enabling users to interact with these systems through graphical interfaces.

Attributes of GUI: AGUI is characterized by

- Windows
- Multiple windows enable variety of information to be prompted on the user's screen at the same time.
- Icons
- may represent files or processes
- Menus
- Command selection possible from a menu instead of typing
- Pointing Device
- Mouse may be used for selecting choices from a menu
- Graphics
- Text and Graphical element can be mixed

Advantages

- Simple to use
- Easy to learn
- Maneuvering within the program and other applications possible (quick switching)
- Full screen is available to be accessed immediately (Fast)

Design Process

A good design is user centered design and requires analyses and understanding of user activities. Design process is given as

- Design prototype (paper based)
- Evaluated by end user (modify if required)
- Design prototype
- Evaluated by end user (modify if required)
- Produce dynamic design prototype
- Evaluated by end user (modify if required)
- Executable prototype
- Evaluated by end user (modify if required)

Design Principles: Design principles to be kept in view are

- Experience and capabilities and needs of the system users must be taken into account during design process of UI
- Designers should be aware of people's physical and mental limitations and should be convinced that people make mistakes

Based on these principals while designing a GUI we need to keep in view

- User familiarity: The interface should use terms and concepts drawn from the experience of the users.
- Consistency i.e. minimal surprise
- Recoverability e.g. undo
- User guidance: help and prompts etc.
- User diversity: Various people with various background may use

User System Interactions: Two issue need to be addressed

How the information from user be provided to the system

How the processed information (from computer) be presented to the user?

Direct Manipulations

Advantages

- User in good control
- Short learning curve
- Immediate feedback

Disadvantages

- May be hard to implement
- Only suitable where there is a visual metaphor for tasks and objects

Video games and CAD Systems are examples.

Menu Selection

Menu Selection

Advantages

- Avoids user error
- Little typing required

Disadvantages

- Slow for experienced users
- Can become complex if many menu options

General purpose softwares

Form fill-in

Advantages

- Simple data entry
- Easy to learn

Disadvantages

- Takes up a lot of screen space

Financial Applications (banks stocks)

Command Language

Advantages are these UI's are powerful and flexible

Disadvantages

- Hard to learn
- Poor error management

Operating systems

Natural Language

Advantages

- Accessible to casual users
- Easily extended

Disadvantages

- Requires more typing

Natural language understanding systems are unreliable

Time tables etc.

Information Presentation: Information presentation means how to present processed information to the users

It may be presented

- Directly (e.g. text):Transformed like in graphical form
- Model-View-Controller approach supports multiple presentations of data
- Static information
  - Adjusted at the start of a session. It remains unchanged during the session
  - May assume numeric or textual value
- Dynamic information
  - Can changes during a session and the changes must be presented to the system user.
  - May assume numeric or textual value
- Display Factors (Information)
  - User requirement
  - Precise information
  - Data relationships
  - Information values change frequency
  - Should user know immediately about the change?
  - Response on change required or not from user
  - Direct or transformed?
  - Textual or numeric?
  - Multiple information presentation
  - Analog and Digital Presentation

Digital presentation is compact so takes lesser screen space and can present precise value while Analog presentation give good information by just getting an overview and present information of outliers etc.

Dynamic Display

- Time
- Temperature
- Pressure

Displaying Relative Values

- Progress bar etc.

- Data Visualization
- Used to display large amount of information
- Reveals trends

Examples

Weather information from a number of sources and Traffic information at toll plazas.

Use of Colors

Extra dimension can be added and can also be used for warning.

Guideline to use Colors

- Too many colors should not be used
- Color code should also be provided and the user may be provided flexibility to change color code.
- Designing must be done for monochrome and colors should be added afterwards
- Color codes should be consistent
- Keep lower resolutions in mind

User Facilitation

The user guidance system (like on – line help, error messages and manuals) should be integrated with the user interface

Animation

Walt Disney created animated objects. Animate mean bring to life

We perceive motion persistence of vision (1/25 seconds). If 25 frames are played in one second are perceived as motion.

Kinematics

Study of movement and motion of the structure that have joints (without consideration of cause of motion)

Animation Types

- Cel Animation: based on changes that occur from one frame to another.
- Path Animation which is created using Key frames by tweening

References:

Fundamentals of Computer Graphics Third Edition by Peter Shirley and Steve Marschner

Interactive Computer Graphics, A Top-down Approach with OpenGL (Sixth Edition) by Edward Angel.

Computer Graphics, Principals and Practice, Foley

Computer Graphics by Hearn and Baker