Unit 4
Graphical User Interface and Interactive Input methods and 3D Concepts

- Introduction to user Dialogue
- Input of Graphical Data
- Input Functions- Introduction to Input functions and input modes, Interactive Picture Construction Techniques
- Virtual -Reality Environments.
- 3D Concepts – Three Dimensional Display Methods,
- Three Dimensional Object Representations – Introduction to different methods used for representation of Three Dimensional Objects.

Interactive Input Methods and Graphical User Interfaces

➔ The User Dialogue

For a particular application the user's model serves as the basis for the design of the dialogue. The user's model describes the following :

- what the system is designed to accomplish
- what graphics operations are available.
- States the type of object that can be displayed
- and How the objects can be manipulated.

All information in the user dialogue is then presented in the language of the application.

General considerations in structuring a user dialogue :

1) Windows and Icons

A window system provides a window – manager interface for the user and functions for handling the display and manipulation of the windows. Common functions for the window system are opening and closing windows, repositioning windows, resizing windows, and display routines that provide interior and exterior clipping and other graphics functions. Windows are displayed with sliders, buttons and menu icons.

Icons representing objects such as furniture items and circuit elements are often referred as application icons. The icons representing actions, such as rotate, scale, clip, and paste are called **control icons or command icons.**

2) Accommodating multiple Skill Level

Usually, interactive graphical interfaces provide several methods for selecting actions. For example, options could be selected by pointing at an icon and clicking different mouse buttons, or by accessing pull down or pop – up menus or by typing keyboard commands. This allows a package to accommodate users that have different skill levels. For less experienced user, an interface with set of menus and options is easy to learn and remember.

Experienced user want speed with more inputs from keyboard. Actions are selected with function keys or combinations of keyboard keys. User will remember shortcuts for commonly used actions.

3) Consistency

An important design consideration in an interface is consistency. For example, a particular icon shape should always have a single meaning, rather than serving to represent different actions or objects depending on the context. It also means always placing menus in the same relative positions so that a user does not have to hunt for a particular options.

Inconsistent model is difficult for a user to understand and do work in an effective way.

4) Minimizing Memorization

Operations in an interface should also be structured so that they are easy to understand and to remember. Obscure, complicated, inconsistent command format lead to confusion and reduction in the effectiveness of the user of package.

Icons and windows systems also aid in minimizing memorization.

5) Backup and error handling

A mechanism for backing up during a sequence of operations is an other common feature of interface. Often an operation can be canceled before execution is completed, with the system restored to the state it was in before the operation was started.

Backup can be provided in many forms. A standard undo key or command is used to cancel a single operation. Sometimes operations can not be undone. Once we have deleted the trash in the desktop wastebasket, we can not recover the deleted files.
Good diagnostics and error messages are designed to help to determine the cause of an error.

6) Feedback

As each input is received, the system normally provides some type of response. An object is highlighted, an icon appears or a message is displayed. This not only informs us that the input has been received, but it also tells us what the system is doing. If processing cannot be completed within a few seconds, several feedback messages might be displayed to keep us informed of the progress of the system.

To speed system response, feedback techniques can be chosen to take advantages of the operating characteristics of type of devices in use,.

Input of Graphics Data

In order to be able to interact with the graphical image input methods are required. These can be used to just change the location and orientation of the camera, or to change specific settings of the rendering itself. Different devices are more suitable for changing some settings then others.

Input methods

Input methods can be classified using the following categories: –
Locator p(x,y)– Stroke – String – Valuator – Choice – Pick

1) Locator : A device that allows the user to specify one coordinate position by selecting screen position. Different methods can be used, such as a mouse cursor, where a location is chosen by clicking a button, or a cursor that is moved using different keys on the keyboard. Touch screens can also be used as locators; the user specifies the location by inducing force onto the desired coordinate on the screen.

Example of locator devices: Mouse, light pen, Joystick, Trackball, Spaceball, thumbwheels, dials, a digitizer stylus or hand cursor.

2) Stroke : A device that allows the user to specify a set of coordinate positions which is equivalent to multiple calls on locator device. The positions can be specified, for example, by dragging the mouse across the screen while a mouse button is kept pressed. On release, a second coordinate can be used to define a rectangular area using the first coordinate in addition.

Example of stroke devices: Mouse, Joystick, Trackball, tablet hand cursor.

3) String: A device that allows the user to specify text input. A text input widget in combination with the keyboard is used to input the text. Also, virtual keyboards displayed on the screen where the characters can be picked using the mouse can be used if keyboards are not available to the application.

4) Valuator: A device that allows the user to input a scalar value. Similar to string inputs, numeric values can be specified using the keyboard. Often, up-down-arrows are added to increase or decrease the current value. Rotary devices, such as wheels can also be used for specifying numerical values. Often times, it is useful to limit the range of the numerical value depending on the value.

5) Choice: A device that allows the user to specify a menu option. Typical choice devices are menus or radio buttons which provide various options the user can choose from. For radio buttons, often only one option can be chosen at a time. Once another option is picked, the previous one gets cleared.

6) Pick: A device that allows the user to specify a component of a picture. Similar to locator devices, a coordinate is specified using the mouse or other cursor input devices and then back-projected into the scene to determine the selected 3-D object. It is often useful to allow a certain "error tolerance" so that an object is picked even though the user did not exactly onto the object but close enough next to it. Also, highlighting objects within the scene can be used to traverse through a list of objects that fulfill the proximity criterion. Certain applications do not allow the use of mouse or keyboard. In particular, 3-D environments, where the user roams freely within the scene, mouse or keyboard would unnecessarily bind the user to a certain location. Other input methods are required in these cases, such as a wireless gamepad or a 3-D stylus, that is tracked to identify its 3-D location.

Example : Data glove

Input Functions

Graphical input functions can be set up to allow users to specify the following options:

- which physical devices are to provide input within a particular logical classification (For example, a tablet used as a stroke device)
- How the graphics program and devices are to interact (input mode). Either the program or the devices can initiate data entry, or both can operate simultaneously.
- When the data are to be input and which device is to be used at that time to deliver a particular input type to the specified data variables.

Input Modes

Input devices contain a trigger which can be used to send a signal to the operating system - Button on mouse - Pressing or releasing a key

•When triggered, input devices return information (their measure) to the system

- Mouse returns position information

- Keyboard returns ASCII code

In addition to multiple types of logical input devices, we can obtain the measure of a device in three distinct modes: 1) Request mode, 2) Sample mode, and 3) Event mode. It defined by the relationship between the measure process and the trigger. Normally, the initialization of an input device starts a measure process.

1) Request mode: In request mode, the application program initiates data entry. Input values are requested and processing is suspended until the required values are received. This input mode corresponds to typical input operation in a general programming language. The program and the input devices operate alternatively. Devices are put into a wait state until an input request is made, then the program waits until the data are delivered.

2) Sample mode: In sample mode, the application program and input devices operate independently. Input devices may be operating at the same time that the program is processing other data. New input values from the input devices are stored, replacing previously input data values. When the program requires new data, it samples the current values from the input devices.

3) Event mode: The previous two modes are not sufficient for handling the variety of possible human-computer interactions that arise in a modern computing environment. The can be done in three steps: 1) Show how event mode can be described as another mode within the measure trigger paradigm. 2) Learn the basics of client-servers when event mode is preferred, and 3) Learn how OpenGL uses GLUT to do this. In an environment with multiple input devices, each with its own trigger and each running a measure process. Each time that a device is triggered, an event is generated. The device measure, with the identifier for the device, is placed in an event queue. The user program executes the events from the queue. When the queue is empty, it will wait until an event appears there to execute it. Another approach is to associate a function called a callback with a specific type of event. This is the approach we are taking.

Interactive picture construction techniques

Interactive picture- construction methods are commonly used in variety of applications, including design and painting packages. These methods provide user with the capability to position objects, to constrain fig. to predefined orientations or alignments, to sketch fig., and to drag objects around the screen. Grids, gravity fields, and rubber band methods are used to aid in positioning and other picture construction operations. The several techniques used for interactive picture construction that are incorporated into graphics packages are:

(1) Basic positioning methods:-coordinate values supplied by locator input are often used with positioning methods to specify a location for displaying an object or a character string. Coordinate positions are selected interactively with a pointing device, usually by positioning the screen cursor.

(2) constraints:-A constraint is a rule for altering input coordinates values to produce a specified orientation or alignment of the displayed coordinates. the most common constraint is a horizontal or vertical alignment of straight lines.

(3) Grids:- Another kind of constraint is a grid of rectangular lines displayed in some part of the screen area. When a grid is used, any input coordinate position is rounded to the nearest intersection of two grid lines.

(4) Gravity field:-When it is needed to connect lines at positions between endpoints, the graphics packages convert any input position near a line to a position on the line. The conversion is accomplished by creating a gravity area around the line. Any related position within the gravity field of line is moved to the nearest position on the line. It illustrated with a shaded boundary around the line.

(5) Rubber Band Methods:-Straight lines can be constructed and positioned using rubber band methods which stretch out a line from a starting position as the screen cursor.

(6) Dragging:-This methods move object into position by dragging them with the screen cursor.

(7) Painting and Drawing:-Cursor drawing options can be provided using standard curve shapes such as circular arcs and splices, or with freehand sketching procedures. Line widths, line styles and other attribute options are also commonly found in painting and drawing packages.

Virtual Reality Environments

In virtual reality environment interactive input is accomplished in this environment with a data glove, which is capable of grasping and moving objects displayed in a virtual scene. The computer generated scene is displayed through a head mounted viewing system as a stereoscopic projection.

Another method for generating virtual scene is to display stereoscopic projections on raster monitor with the two stereoscopic views displayed on alternate refresh cycles. The scene is then viewed through stereoscopic glasses. Interactive object manipulations can again be accomplished with a data glove and a tracking device to monitor the glove position and orientation relative to the position of object in the scene.

THREE DIMENSIONAL DISPLAY METHODS

To obtain display of a three-dimensional scene that has been modeled in world coordinates. we must first set up a coordinate reference for the "camera". This coordinate reference defines the position and orientation for the plane of the camera film which is the plane we want to us to display a view of the objects in the scene. Object descriptions are then transferred to the camera reference coordinates and projected onto the selected display plane. We can then display the objects in wireframe (outline) form, or we can apply lighting surface rendering techniques to shade the visible surfaces.

1) Parallel Projection

In a parallel projection, parallel lines in the world-coordinate scene projected into parallel lines on the two-dimensional display plane. One method for generating a view of a solid object is to project points on the object surface along parallel lines onto the display plane. By selecting different viewing positions, we can project visible points on the object onto the display plane to obtain different two dimensional views of the object. This technique is used in engineering and architectural drawings to represent an object with a set of views that maintain relative proportions of the object.

Top, Left, Right, Front view of the objects : based on position of camera – viewing position.

2) Perspective Projection

Another method for generating a view of a three-dimensional scene is to project points to the display plane along converging paths. This causes objects farther from the viewing position to be displayed smaller than objects of the same size that are nearer to the viewing position. In a perspective projection, parallel lines in a scene that are not parallel to the display plane are projected into converging lines. Scenes displayed using perspective projections appear more realistic, since this is the way that our eyes and a camera lens form images. In the Perspective Projection view, parallel lines appear to converge to a distant point in the background, and distant objects appear smaller than objects closer to the viewing position.

3) Depth Cueing

Depth information is important so that we can easily identify, for a particular viewing direction, which is the front and which is the back of displayed objects. There are several ways in which we can include depth information in the two-dimensional representation of solid objects.

A simple method for indicating depth with wireframe displays is to vary the intensity of objects according to their distance from the viewing position.

Application of depth cueing is modeling the effect of the atmosphere on the perceived intensity of objects. More distant objects appear dimmer to us than nearer objects due to light scattering by dust particles, haze, and smoke. Some atmospheric effects can change the perceived color of an object, and we can model these effects with depth cueing.

4) Visible Line and Surface Identification

We can also clarify depth relation ships in a wireframe display by identifying visible lines in some way. The simplest method is to highlight the visible lines or to display them in a different color. Another technique, commonly used for engineering drawings, is to display the nonvisible lines as dashed lines. Another approach is to simply remove the nonvisible lines.

5) Surface Rendering

Added realism is attained in displays by setting the surface intensity of objects according to the lighting conditions in the scene and according to assigned surface characteristics. Lighting specifications include the intensity and positions of light sources and the general background illumination required for a scene. Surface properties of objects include degree of transparency and how rough or smooth the surfaces are to be. Procedures can then be applied to generate the correct illumination and shadow regions for the scene. Surface-rendering methods are combined with perspective and visible-surface identification to generate a degree of realism in a displayed scene.

6) Exploded and Cutaway View

Many graphics packages allow object to be defined as hierarchical structures, so that internal details can be stored. Exploded and cutaway views of such objects can then be used to show the internal structure and relationship of the object parts.

7) Three-Dimensional and Stereoscopic View

Three-dimensional views can be obtained by reflecting a raster image from a vibrating flexible mirror. The vibrations of the mirror are synchronized with the display of the scene on the CRT. As the mirror vibrates, the focal length varies so that each point in the scene is projected to a position corresponding to its depth.

Stereoscopic devices present two views of a scene: one for the left eye and the other for the right eye.

Three Dimensional Object Representations

Polygon Surfaces

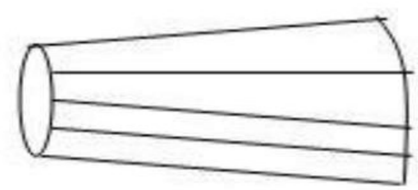
Objects are represented as a collection of surfaces. 3D object representation is divided into two categories.

Boundary Representations B−reps − It describes a 3D object as a set of surfaces that separates the object interior from the environment.

Space–partitioning representations − It is used to describe interior properties, by partitioning the spatial region containing an object into a set of small, non-overlapping, contiguous solids usually cubes.

The most commonly used boundary representation for a 3D graphics object is a set of surface polygons that enclose the object interior. Many graphics system use this method. Set of polygons are stored for object description. This simplifies and speeds up the surface rendering and display of object since all surfaces can be described with linear equations.

The polygon surfaces are common in design and solid-modeling applications, since their **wireframe display** can be done quickly to give general indication of surface structure. Then realistic scenes are produced by interpolating shading patterns across polygon surface to illuminate.

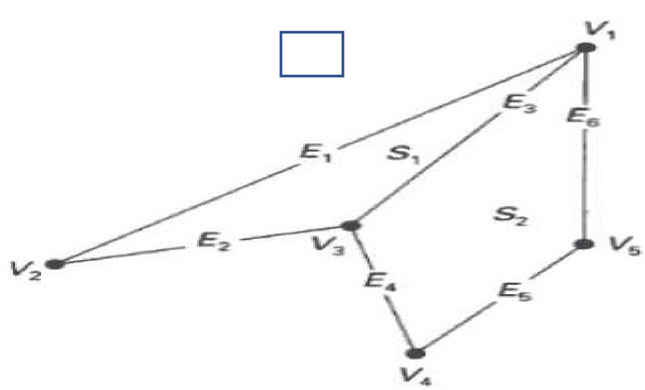


A 3D object represented by polygons

Polygon Tables

In this method, the surface is specified by the set of vertex coordinates and associated attributes. As shown in the following figure, there are five vertices, from v_1 to v_5.

- Each vertex stores x, y, and z coordinate information which is represented in the table as v_1: x_1, y_1, z_1. (3D plane)
- The Edge table is used to store the edge information of polygon. In the following figure, edge E_1 lies between vertex v_1 and v_2 which is represented in the table as E_1: v_1, v_2.
- Polygon surface table stores the number of surfaces present in the polygon. From the following figure, surface S_1 is covered by edges E_1, E_2 and E_3 which can be represented in the polygon surface table as S_1: E_1, E_2, and E_3.



Plane Equations

VERTEX TABLE	
V_1:	x_1, y_1, z_1
V_2:	x_2, y_2, z_2
V_3:	x_3, y_3, z_3
V_4:	x_4, y_4, z_4
V_5:	x_5, y_5, z_5

EDGE TABLE	
E_1:	V_1, V_2
E_2:	V_2, V_3
E_3:	V_3, V_1
E_4:	V_3, V_4
E_5:	V_4, V_5
E_6:	V_5, V_1

POLYGON-SURFACE TABLE	
S_1:	E_1, E_2, E_3
S_2:	E_3, E_4, E_5, E_6

The equation for plane surface can be expressed as −

$$Ax + By + Cz + D = 0$$

Where x,y,z is any point on the plane, and the coefficients A, B, C, and D are constants describing the spatial properties of the plane. We can obtain the values of A, B, C, and D by solving a set of three plane equations using the coordinate values for three non collinear points in the plane. Let us assume that three vertices of the plane are (x_1, y_1, z_1), (x_2, y_2, z_2) and (x_3, y_3, z_3).

Let us solve the following simultaneous equations for ratios A/D, B/D, and C/D. You get the values of A, B, C, and D.

$$A/D \ x_1 + B/D \ y_1 + C/D \ z_1 = -1$$

$$A/D \ x_2 + B/D \ y_2 + C/D \ z_2 = -1$$

$$A/D \ x_3 + B/D \ y_3 + C/D \ z_3 = -1$$

To obtain the above equations in determinant form, apply Cramer's rule to the above equations.

$$A = \begin{bmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{bmatrix} B = \begin{bmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{bmatrix} C = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix} D = - \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix}$$

For any point x,y,z with parameters A, B, C, and D, we can say that −

- $Ax + By + Cz + D \neq 0$ means the point is not on the plane.
- $Ax + By + Cz + D < 0$ means the point is inside the surface.
- $Ax + By + Cz + D > 0$ means the point is outside the surface.

Polygon Meshes

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.

Blobby Object:

Some objects do not maintain a fixed shape, but change their surface characteristics in certain motions or when in proximity (nearness) to other objects. Examples in this class of objects include molecular structure, water droplets and other liquid effects, melting objects and muscle shapes in the human body. These objects can be described as exhibiting “blobbiness” and are often simply referred as objects, since their shapes show a certain degree fluidity.

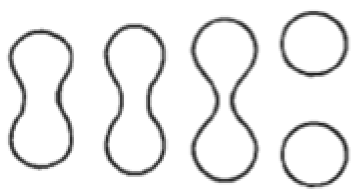


Figure 10-14
Molecular bonding. As two molecules move away from each other, the surface shapes stretch, snap, and finally contract into spheres.

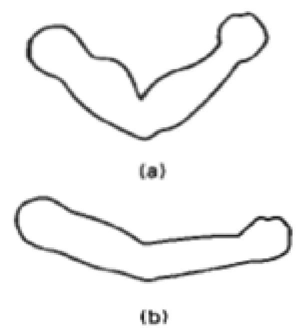


Figure 10-15
Blobby muscle shapes in a human arm.

Spline Representations

In drafting terminology, a spline is a flexible strip used to produce a smooth curve through a designated set of points. Several small weights are distributed along the length of the strip to hold it in position on the drafting table as the curve is drawn. The **spline curve** originally referred to a curve drawn in this manner. We can mathematically describe such a curve with a piecewise cubic. Polynomial function whose first and second derivatives are continuous across the various curve section.

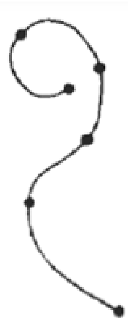


Figure 10-19
A set of six control points interpolated with piecewise continuous polynomial sections.

In Computer graphics, the term **spline curve** refers to any composite curve formed with polynomial sections satisfying specified continuity conditions at the boundary of the pieces.

A spline surface can be described with two sets of orthogonal spline curves.

Figure 10-20
A set of six control points approximated with piecewise continuous polynomial sections

Splines are used in graphics applications to design curve and surface shapes, to digitize drawings, for computer storage and to specify animation paths for the objects or the camera in a scene.