

Computer Graphics

Lecture 15

Two Dimensional Viewing

We now consider the formal mechanism for displaying views of a picture on an output device. Typically, a graphics package allows a user to specify which part of a defined picture is to be displayed and where that part is to be placed on the display device. Any convenient Cartesian coordinate system, referred to as the world-coordinate reference frame, can be used to define the picture. For a two-dimensional picture, a view is selected by specifying a subarea of the total picture area. A user can select a single area for display, or several areas could be selected for simultaneous display or for an animated panning sequence across a scene. The picture parts within the selected areas are then mapped onto specified areas of the device coordinates. When multiple view areas are selected, these areas can be placed in separate display locations, or some areas could be inserted into other, larger display areas. Transformations from world to device coordinates involve translation, rotation, and scaling operations, as well as procedures for deleting those parts of the picture that are outside the limits of a selected display area.

The Viewing Pipeline

A world-coordinate area selected for display is called a window. An area on a display device to which a window is mapped is called a viewport. The window defines what is to be viewed; the viewport defines where it is to be displayed. Often, windows and viewports are rectangles in standard position, with the rectangle edges parallel to the coordinate axes. Other window or viewport geometries, such as general polygon shapes and circles, are used in some applications, but these shapes take longer to process. In general, the mapping of a part of a world-coordinate scene to device coordinates is referred to as a viewing transformation. Sometimes the two-dimensional viewing transformation is simply referred to as the window-to-viewport transformation or the windowing transformation. But, in general, viewing involves more than just the transformation from the window to the viewport. Figure 6-1 illustrates the mapping of a picture section that falls within a rectangular window onto a designated rectangular viewport.

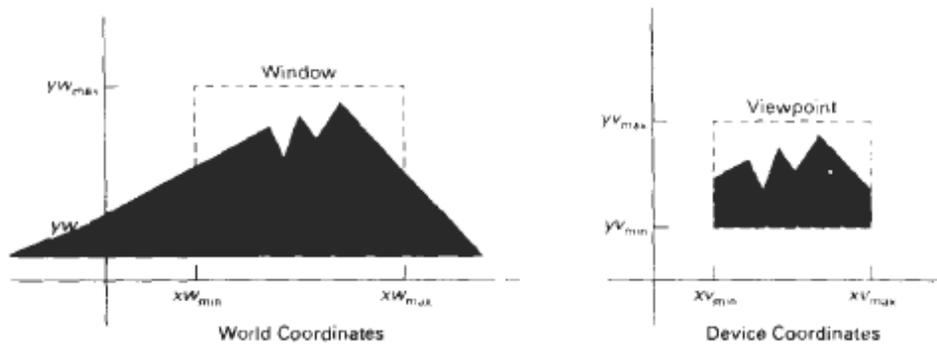


Figure 6-1
A viewing transformation using standard rectangles for the window and viewport.

In computer graphics terminology, the term window originally referred to an area of a picture that is selected for viewing, as defined at the beginning of this section. Unfortunately, the same term is now used in window-manager systems to refer to any rectangular screen area that can be moved about, resized, and made active or inactive.

Some graphics packages that provide window and viewport operations allow only standard rectangles, but a more general approach is to allow the rectangular window to have any orientation. In this case, we carry out the viewing transformation in several steps, as indicated in Fig. 6-2. First, we construct the scene in world coordinates using the output primitives and attributes previously. Next to obtain a particular orientation for the window, we can set up a two-dimensional **viewing-coordinate system** in the world-coordinate plane, and define a window in the viewing coordinate system. The viewing-coordinate reference frame is used to provide a method for setting up arbitrary orientations for rectangular windows. Once the viewing reference frame is established, we can transform descriptions in world coordinates to viewing coordinates. We then define a viewport in normalized coordinates (in the range from 0 to 1) and map the viewing-coordinate description of the scene to normalized coordinates. At the final step all parts of the picture that lie outside the viewport are clipped, and the contents of the viewport are transferred to device coordinates. Figure 6-3 illustrates a rotated viewing-coordinate reference frame and the mapping to normalized coordinates.

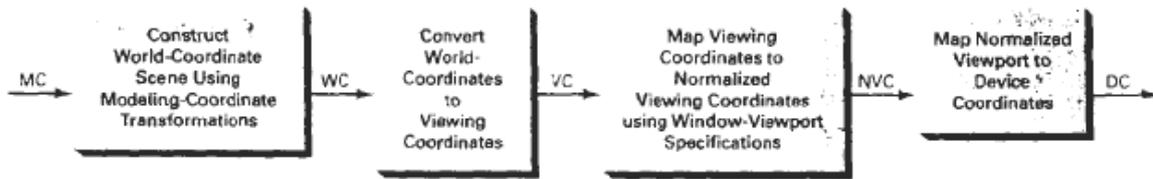


Figure 6-2
The two-dimensional viewing-transformation pipeline.

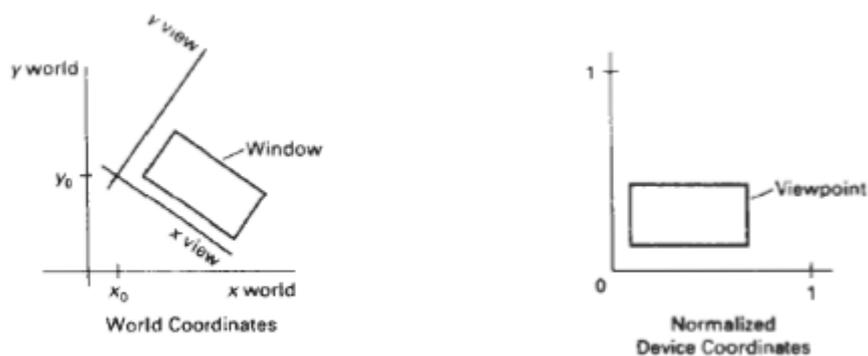


Figure 6-3
Setting up a rotated world window in viewing coordinates and the corresponding normalized-coordinate viewport.

Clipping Operations

Generally, any procedure that identifies those portions of a picture that are either inside or outside of a specified region of space is referred to as a **clipping algorithm**, or simply **clipping**. The region against which an object is to be clipped is called a clip window.

Applications of clipping include extracting part of a defined scene for viewing; identifying visible surfaces in three-dimensional views; antialiasing line segments or object boundaries; creating objects using solid-modelling procedures; displaying a multi-window environment; and drawing and painting operations that allow parts of a picture to be selected for copying, moving, erasing, or duplicating. Depending on the application, the clip window can be a general polygon or it can even have curved boundaries.

For the viewing transformation, we want to display only those picture parts that are within the window area (assuming that the clipping flags have not been set to no-clip). Everything outside the window is discarded. Clipping algorithms can be applied in world coordinates, so that only the contents

of the window interior are mapped to device coordinates. Alternatively, the complete world-coordinate picture can be mapped first to device coordinates, or normalized device coordinates, then clipped against the viewport boundaries. World-coordinate clipping removes those primitives outside the window from further consideration, thus eliminating the processing necessary to transform those primitives to device space. Viewport clipping, on the other hand, can reduce calculations by allowing concatenation of viewing and geometric transformation matrices. But viewport clipping does require that the transformation to device coordinates performed for all objects, including those outside the window area. On raster systems, clipping algorithms are often combined with scan conversion

Clipping Types

Point Clipping

Assuming that the clip window is a rectangle in standard position, we save a point $P = (x, y)$ for display if the following inequalities are satisfied:

$$\begin{aligned}xw_{\min} \leq x \leq xw_{\max} \\yw_{\min} \leq y \leq yw_{\max}\end{aligned}\tag{6-5}$$

where the edges of the clip window (xw_{\min} , xw_{\max} , yw_{\min} , yw_{\max}) can be either the world-coordinate window boundaries or viewport boundaries. If any one of these four inequalities is not satisfied, the point is clipped (not saved for display).

Although point clipping is applied less often than line or polygon clipping, some applications may require a point clipping procedure. For example, point clipping can be applied to scenes involving explosions or sea foam that are modelled with particles (points) distributed in some region of the scene.