Graphics Systems and Model

Raster Images: Raster image is an image of a 2-dimensional array of square (or generally rectangular) cells called *pixels* (short for "picture elements"). Such images are sometimes called *pixel maps*.

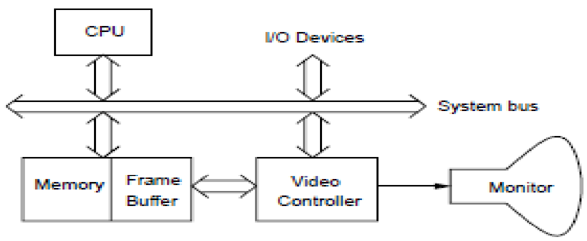
The simplest example is an image made up of black and white pixels, each represented by a single bit (0 for black and 1 for white). This is called a *bitmap*. For gray-scale (or *monochrome*) raster images raster images, each pixel is represented by assigning it a numerical value over some range (e.g., from 0 to 255, ranging from black to white). There are many possible ways of encoding color images.

Graphics Devices: The standard interactive graphics device is called a *raster display*. As with a television, the display consists of a two-dimensional array of pixels. There are two common types of raster displays.

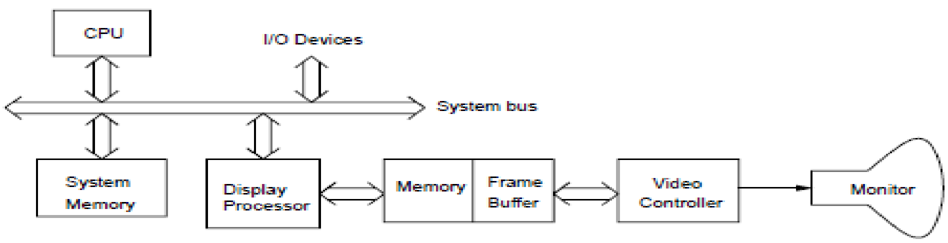
Video displays: consist of a screen with a phosphor coating, that allows each pixel to be illuminated momentarily when struck by an electron beam. A pixel is either illuminated (white) or not (black). The level of intensity can be varied to achieve arbitrary gray values. Because the phosphor only holds its color briefly, the image is repeatedly rescanned, at a rate of at least 30 times per second.

Liquid crystal displays (LCD's): use an electronic field to alter polarization of crystalline molecules in each pixel. The light shining through the pixel is already polarized in some direction. By changing the polarization of the pixel, it is possible to vary the amount of light which shines through, thus controlling its intensity.

Irrespective of the display hardware, the computer program stores the image in a two-dimensional array in RAM of pixel values (called a *frame buffer*). The display hardware produces the image line-by-line (called *raster lines*). A hardware device called a *video controller* constantly reads the frame buffer and produces the image on the display. The frame buffer is not a device. It is simply a chunk of RAM memory that has been allocated for this purpose. A program modifies the display by writing into the frame buffer, and thus instantly altering the image that is displayed. An example of this type of configuration is shown below.



Simple Raster Graphics System



Raster Graphics with Display Processor

More sophisticated graphics systems, which are becoming increasingly common these days, achieve great speed by providing separate hardware support, in the form of a *display processor* (more commonly known as a *graphics accelerator* or *graphics card* to PC users). This relieves the computer's main processor from much of the mundane repetitive effort involved in maintaining the frame buffer. A typical display processor will provide assistance for a number of operations including the following:

Transformations: Rotations and scalings used for moving objects and the viewer's location.

Clipping: Removing elements that lie outside the viewing window.

Projection: Applying the appropriate perspective transformations.

Shading and Coloring: The color of a pixel may be altered by increasing its brightness. Simple shading involves smooth blending between some given values. Modern graphics cards support more complex procedural shading.

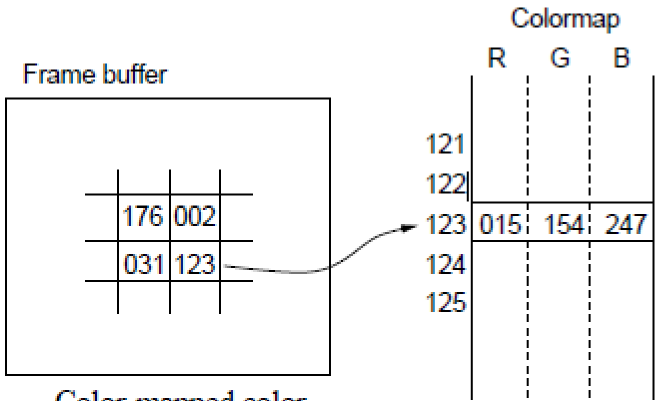
Texturing: Coloring objects by "painting" textures onto their surface. Textures may be generated by images or by procedures.

Hidden-surface elimination: Determines which of the various objects that project to the same pixel is closest to the viewer and hence is displayed.

Color: The method chosen for representing color depends on the characteristics of the graphics output device (e.g.,whether it is *additive* as are video displays or *subtractive* as are printers). It also depends on the number of bits per pixel that are provided, called the *pixel depth*. For example, the most method used currently in video and color LCD displays is a *24-bit RGB* representation. Each pixel is represented as a mixture of red, green and blue components, and each of these three colors is represented as a 8-bit quantity (0 for black and 255 for the brightest color).

In many graphics systems it is common to add a fourth component, sometimes called *alpha*, denoted A. This component is used to achieve various special effects, most commonly in describing how opaque a color is. In some instances 24-bits may be unacceptably large. For example, when downloading images from the web, 24-bits of information for each pixel may be more than what is needed. A common alternative is to used a *color map*, also called a *color look-up-table* (LUT). (This is the method used in most gif files, for example.) In a typical instance, each pixel is represented by an 8-bit quantity in the range from 0 to 255. This number is an index to a 256-element array, each of whose entries is a 234-bit RGB value. To represent the image, we store both the LUT and the image itself. The 256 different colors are usually chosen so as to produce the best possible reproduction of the image. For example, if the image is mostly blue and red, the LUT will contain many more blue and red shades than others.

A typical photorealistic image contains many more than 256 colors. This can be overcome by a fair amount of clever trickery to fool the eye into seeing many shades of colors where only a small number of distinct colors exist. This process is called *digital halftoning*. Colors are approximated by putting combinations of similar colors in the same area. The human eye averages them out.



Color-mapped color.