Graphics & Image Data Representations

- Graphics/image data-type
- Popular file formats.

1-Bit images: Images consists of pixels - Picture element in digital images. A 1-bit image consists of on & off bits only & thus is the simple type of image. Each pixel is stored as a single bit (0/1). Hence, such an image is also refered as binary image.

It is also called, a 1-bit monochrome image, since it contains no color. A 640×480 monochrome image requires 38.4 Kbytes of storage.

8-bit Gray-Level images: This is used for each pixel has a gray value b/w 0 & 255. Each pixel is represented by a single byte - Ex: a dark pixel might have a value of 10 & a bright one might be 230.

The entire image can be thought of as a two dimensional array of pixel values. we refer to such an array as a bitmap

Image resolution refers to the number of pixels in a digital image. high resolution for such an image might be 1600×1200 whereas lower resolution might be 640×480.

Such an array must be stored in hardware we call this hardware a frame buffer. Special hardware called a video card. The resolution of the video card does not have to match the desired resolution of the image, but if not enough video card memory is available, the data has to be shifted around in RAM for display.

we can think of the 8-bit image as a set of 1-bit bitplanes, where each plane consists of a 1-bit representation of the image at higher & higher levels of elevation: a bit is turned on it the image pixel has a nonzero value at or above that bit level.

Ex:

Each pixel is usually stored as a byte (0 to 255) so 640 grayscale image requires 300 kilobytes of storage

In case of 600 dot-per-inch (dpi) laser printer, such a device can usually only print a pot or not print it, However 600x600 image could be printed in a 1-inch space.

Image Data Type: This section introduce some of the most common data types for graphics & image file formats: 24-bit color & 8 bit colors. some formats are restricted to particular hardware/os platform, while others are platform independent, or cross-platform formats. Some formats are not cross-platforms, conversion applications can recognize & translate formats from one system to another.

24Bit color images

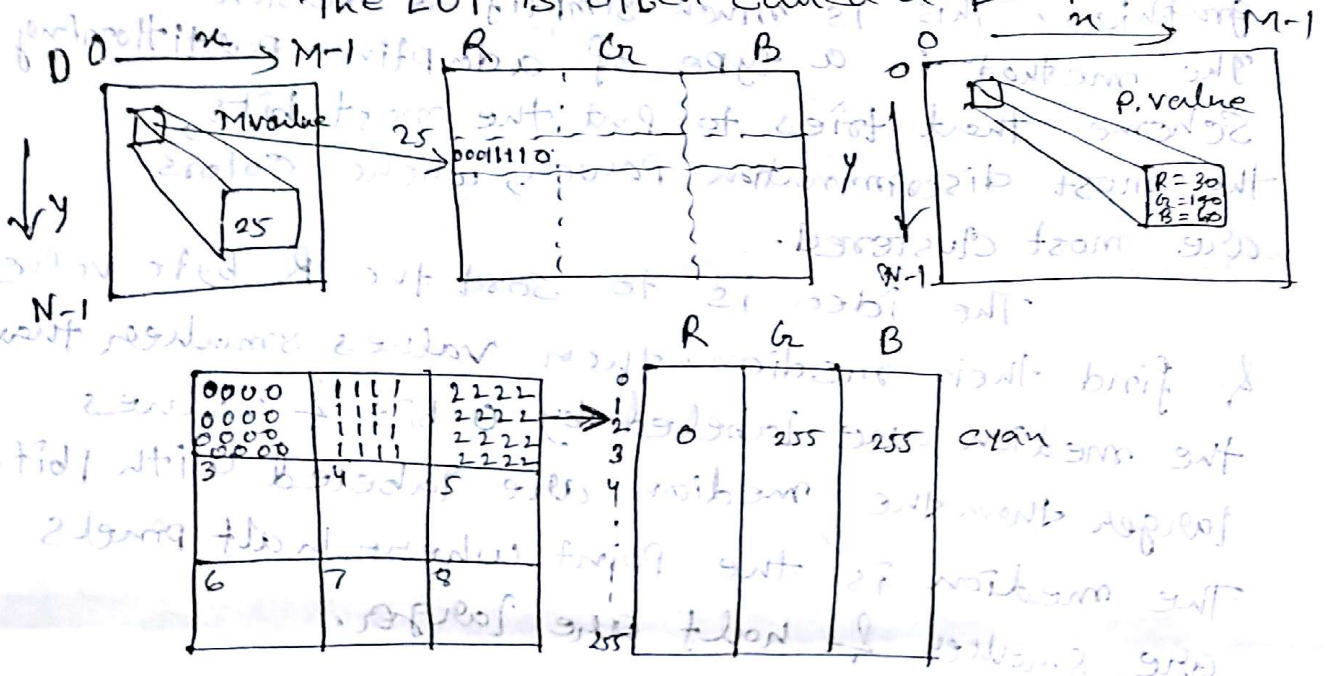
In a color 24-bit image, each pixel is represented by three bytes, usually representing RGB. Since each value is in the range 0-255, this format supports 256 X 256 X 256. a 640 x 480, 24 bit color image would require 921.6 kilobytes of storage without any compression.

8-bit color images: Many systems can make use of only 8 bit of color information in producing a screen image. Basically image stores not color but instead just a set of bytes, each of which is an index into a table with 3-byte values that specify the color for a pixel with that lookup table index.

Color Lookup Tables (LUTs)

The idea used in 8 bit color images is to store only the index or code value, for each pixel. Then if a pixel stores, say the value 25, the meaning is to go to row 25 in a color lookup table (LUT)

while images are displayes as two dimensional arrays of values, they are usually stored in row column order as simply a long series of values. For an 8 bit image the image file can store in the file header information just what 8 bit values for R, G, & B correspond to each index. The LUT is often called a palette.



A color picker consists of an array of fairly blocks of color ~~each other~~. In reality a color picker displays the palette colors associated with index values from 0 to 255, Ex: If the user selects the color block with index value 2, then the color meant is cyan with RGB values (0, 255, 255).

A simple animation process is possible via simply changing the color table, this is called color cycling or Palette animation.

Since humans are more sensitive to R & G than to B, we could shrink the R range & G range 0...255 into the 3 bit range 0...7 & shrink the B range down to 2 bit range 0...3, making a total of 8 bits. To shrink R & G we could simply divide the R or G byte value by (256/8) = 32 then truncate. Then each pixel in the image gets replaced by its 8-bit index, & the color LUT serves to generate 24 bit color.

Disadvantage: A slight change in RGB results in shifting to a new code.

A simple alternate soln for this color reduction problem called the median-cut-algo does a better job. This approach derives from computer graphics. This is much simplified version. The method is a type of adaptive partitioning scheme that tries to put the most bits, the most discrimination power, where colors are most clustered.

The idea is to sort the R byte values & find their median. Then values smaller than the median are labeled by 0 bit & values larger than the median are labeled with 1 bit. The median is the point where half pixels are smaller & half are larger.

median cut algo

1. Find the smallest box that contain all the colors in the image.
2. Sort the enclosed colors along the longest dimension of the box.
3. Split the box into two regions at the median of the sorted list.
4. Repeat the above process in steps (2) & (3) until the original color space has been divided into say 256 regions.
5. For every box, call the mean of R, G, & B in that box the representative color for the box.
6. Based on euclidean distance b/w a pixel RGB value & the box centers, assign every pixel to one of the representative color. Replace the pixel by the code in a look up table that indexes representative colors.

Popular File Formats: Some popular file formats for information exchange are described below.

(1) **8 bit GIF format:** Graphics Interchange format is limited to 8-bit (256) color image only. While this produces acceptable color, it is best suited for images with few distinctive colors.

GIF image format has few interesting features. The standard supports interlacing – the successive display of pixels in widely spaced rows by a four-pass display process.

GIF File format

The GIF signature is 6 bytes, The screen descriptor is 7 bytes. The GIF 87 file can contain more than one image definition, usually to fit on several different parts of the screen. each image can contain its own color lookup table, a Local color map, for mapping 8 bits into 24 bit RGB values. The screen descriptor comprises a set of attributes that belong to every image in the file.

According to the GIF87 standard, screen width is 2 bytes, screen Height is the next 2 bytes.

Bits 7 6 5 4 3 2 1 0		
Screen width	2	Raster width in Pixels (LSB first)
Screen height	3, 4	Raster height in Pixels (LSB first)
m | cr | 0 | pixel	5	
Background	6	Background = color index of screen
0'00000000	7	

m=1
cr+1 # bits of color resolution.
Pixel+1 # bits/pixel in image.

Bits 7 6 5 4 3 2 1 0	Byte #	
Red intensity	1	Red value for color index 0
Green "	2	Green " " " " 0
Blue "	3	Blue " " " " 0
Red "	4	Red " " " " 1
⋮	5	Green " " " " 1
⋮	6	

color map (The actual length of the table equals $2^{Pixel+1}$)

7 6 5 4 3 2 1 0		
0 0 1 0 1 1 0 0	1	Image Separator Character
image left	2 3	Start of image in Pixels from the Left side of the screen, (LSB first)
image top	4 5	Start of image in Pixels from the top of the screen (LSB first)
image width	6 7	Width of the image in pixels (LSB first)
image height	8 9	Height of the image in pixels (LSB first)
m | i | 0 0 0 | Pixel	10	m=0 Use global colormap, ignore Pixel m=1 Local color map follows use Pixel i=0 Image formatted in sequential order. i=1 Image formatted in interlaced order. Pixel+1 # bits/Pixel for this image.

GIF image descriptor

Image row	Pass 1	Pass 2	Pass 3	Pass 4	Result

• **JPEG:** Now a days the most important current standard for image compression is JPEG.

The human vision system has some specific limitations, which JPEG takes advantage of to achieve high rates of compression. The eye-brain system cannot see extremely fine detail. If many changes occur within a few pixels, we refer to that image segment as having high spatial frequency ~~spatial frequency~~. That is a great deal of change in (x,y) space. This limitation is even more conspicuous for color vision than for grayscale. Therefore color information in JPEG is decimated (partially dropped or averaged) & then small blocks of an image are represented in the spatial frequency domain (u,v) rather than in (x,y). That is the speed of changes in x & y is evaluated, from low to high & a new image is formed by grouping the coefficients or weights of these speeds.

Weights that correspond to slow changes are then favored, using a simple trick: values are divided by some large integer & truncated. In this way small values are zeroed out. Then a scheme for representing long runs of zeros efficiently is applied, the image is greatly compressed.

Since we effectively throw away a lot of information by the division & truncation step, this compression scheme is "lossy". It is straightforward method, allow the user to choose how large a denominator to use & hence how much information to discard.

PNG: Portable network Graphics. Special features of PNG files include support for up to 48 bits of color information. a large increase. File may also contain gamma-correction information for correct display of color images & alpha channel information for such uses as control of transparency.

Gamma correction: The RGB numbers in an image file are converted back to analog & drive the electron guns in the (CRT). Electrons are emitted proportional to the driving voltage, & we would like to have the CRT system produce light linearly related to the voltage.

The light emitted is actually roughly proportional to the driving voltage, raised to a power, this power is called gamma, with symbol γ.

Thus if the file value in the red channel is R the screen emits light proportional to R^{γ}.

The mechanics of television receiver are the same as those for a computer CRT, Tv systems precorrect for this situation by applying the inverse transformation before transmitting Tv voltage signals.

$$R \to R^{\frac{1}{\gamma}} \Rightarrow (R^{\frac{1}{\gamma}})^{\gamma} = R.$$

we arrive at linear signals.

light o/p with no gamma correction, at that case we see that darker values are displayed too dark.

In case of gamma correction by $R^{\frac{1}{\gamma}}$ where voltage normalized to the range 0 to 1,