Digital Media SIG

December, 2011
Tom Burt

News Items:

- DVDFab Products – http://dvdfab.com
- FireFox – growing discontent, financial issues w Google
- New “4K” Ultra HDTV’s on the horizon for home use 4096 x 2160, 36-bit color

Round the room Q & A:

Feature Presentation

Inside a Digital Image

We take digital images for granted. Used in camera’s, scanned images and documents, videos, digital TV.

This session will look at various aspects of the digital image:

- How it’s represented as a computer data file
- How the digital image is rendered onto various display and print devices
- How digital images are captured

To help illustrate some of the key concepts, we’ll use the open source Paint.NET image editing tool.

Today, we’ll work with Windows Bitmap (.bmp) files. These are simple, uncompressed image files. All other formats, like .jpg, .gif, .png basically are compressed representations of bitmaps.

A digital image is an organized data structure representing rows and columns of numbers, each representing one individual dot in the image.

Each individual dot is called a Picture Element or Pixel for short.

Sizes of images can vary. A typical image on a computer screen might have 1024 horizontal rows each with 768 vertical columns of dots. This is usual expressed as a resolution of 1024 x 768. A standard modern HDTV has a resolution of 1920 x 1080 dot. Many digital cameras can capture images with resolutions of 4000 x 2000 or even higher.

In an image file, each individual pixel can be represented by one byte (8 bits = 256 colors or grey scales), two bytes (16 bits = 65,536 colors), 3 bytes (24 bits = 16,777,216 colors) or 4 bytes (30 bits = 1,073,741,824 colors). As the pixels get wider, the "color depth" increases. 24 bit color is the commonest today. 30 bit color is available in some HDTVs.

Let's look at a 24 bit (3 byte) pixel.

The pixel is divided up into 3 sub-pixels of one byte each – one for red, one for green and one for blue. The numeric value of each subpixel can vary from 0 to 255, representing the intensity of that color within the pixel. So for example, a pixel with sub-pixel values of red:128  , green:96  , blue:192  would appear as  when displayed on a screen.

The human eye perceives red, green and blue and the brain's visual cortex combines those primary colors to produce the full spectrum.

A pixel with sub-pixel values of red:0, green:0, blue:0 will appear as pure black.

A pixel with sub-pixel values of red:255, green:255, blue:255 will appear as pure white.

A pixel with sub-pixel values of red:255, green:255, blue:0 will appear as

A pixel with sub-pixel values of red:255, green:0, blue:255 will appear as 

A pixel with sub-pixel values of red:0, green:255, blue:255 will appear as 

Let's examine a real image with Paint.NET …

When viewing an image on a screen, the finer the resolution (more pixels per inch), the less "jaggy" the image appears. When we use the Paint.NET zoom tool, we can examine individual pixels of the image and use the color picker to determine exactly what the red, green, blue values are.

Paint.NET also tells us the current size of the image in pixels (1024 x 768) in this case.

Paint.NET lets you can perform all the classic transformations of the image: rotate, resize, select and crop. (demo)

Paint.NET provides a wide variety of adjustments, including brightness and contrast (demo).

What would a computer program to increase the brightness of an image by 10% look like? (We'll use VB.Net syntax)

```
For i = 0 to 499     ' loop through 500 rows
       For j = 0 to 799     ' loop through 800 columns
              aPixel = Image.GetPixel(i, j) ' Get the pixel at i, j
              aPixel.Red = cByte(min((cSng(aPixel.Red) * 1.1), 255.))
              aPixel.Green = cByte(min((cSng(aPixel.Green) * 1.1), 255.))
              aPixel.Blue = cByte(min((cSng(aPixel.Blue) * 1.1), 255.))
              Image.SetPixel(i, j, aPixel)
       Next j
Next i
```

Let's now look at how an image is displayed on a computer or HDTV screen:

www.howstuffworks.com

The LCD screen is organized into rows and columns of pixels, just like the bitmap image file is. Each pixel is divided into 3 sub-pixels (red, green and blue). Each sub-pixel is controlled by a transistor that can make the liquid crystal completely opaque to light (value 0), transparent to light (value 255) or anywhere in between. A control circuit scans the image in the monitor's display buffer moving through each column in each row and then sets the transparency for each sub-pixel at that pixel position. The entire image is scanned and rendered 60 or 120 times each second. The LCD panel has a back light (either fluorescent or LED) that shines through each of the sub-pixels in the screens.