

Lecture #2: Digital Images

CS106E Spring 2018, Young

In this lecture we will see how computers display images. We'll find out how computers generate color and discover that color on computers works differently than it does for paintings and the printed page. We'll take a look at how computers store images and learn when to use each of the most common image file formats used on the web.

Along the way, we'll learn about compression—an important technique that is used to reduce the amount of space taken by a file. We'll see that compression techniques may be either lossy or lossless.

If anyone is planning to buy a new television anytime soon, we'll take a quick look at what makes 4K HDR televisions special.

How a Computer Display Works

- If we inspect a computer display, we'll discover that it's a giant grid
- For black and white computer displays
 - each element in the grid corresponds to a single bit
 - the corresponding grid element will be black or white depending on whether the bit is a 1 or a 0
 - by turning grid elements on or off we form an image
- We refer to each individual grid element as a picture element or **pixel**.
- The number of pixels going from left-to-right and the number of pixels going from top-to-bottom determine the **screen resolution**.
- The ratio of the width of the screen compared to the height of the screen determine the screen's **aspect ratio**
 - Traditional Screen Resolutions typically had aspect ratios of 4:3 and included
 - 640x480, 1024x768, 1600x1200
 - Widescreen Resolutions are 16:9 and include
 - 1280x1080, 1920x1200, 2560x1440
 - New **4K Televisions** have a 16:9 Aspect Ratio and their resolution is:
 - 3840x2160
- Another important quality in a computer display is how fine-grained the individual pixels are.
 - If the pixels are very small and tightly packed, the human eye won't notice the individual pixels, and display of text, lines, and geometric shapes will look smooth and continuous.







- In contrast, if the screen has large pixels, the output will look blocky, and it will be very obvious that letters, lines, and other display items are composed of pixels.
- This quality is measured in **pixels-per-inch (PPI)**. The higher the PPI, the more finely grained the pixels are and the better items will appear in your monitor.
 - Some monitor specifications provide the pixel pitch (also referred to as dot pitch) rather than PPI. This is the distance between each pixel and correlates directly with PPI.
- A similar measurement of printer quality is the **dots-per-inch (DPI)**.¹ Printouts from high DPI printers will look much nicer than printouts from low DPI printers.

How a Computer Display Shows Color




- If we look at our computer display with a magnifying glass, we will discover that each pixel is actually composed of three different bars—a red bar, a green bar, and a blue bar.
- By changing the amount of Red, Green, and Blue (abbreviated as **RGB**) we can create different colors.
- Just as the screen resolution determines the number of pixels on the screen, the **color depth** determines how many colors can be displayed.
 - The color depth is the number of bits set aside to represent each pixel. More bits per pixel means we can represent more colors on our screen.
 - Early computers and video game consoles had very limited ability to display colors. Color depths used by these included
 - black-and-white also referred to as *monochrome*
 - 8-bit *grayscale* allowing $2^8 = 256$ variations of gray, but no color at all
 - 2-bit color (displaying up to 4 colors as $2^2 = 4$)
 - 4-bit color (displaying up to 16 colors as $2^4 = 16$)
 - 8-bit color allowing us to display 256 colors
 - modern computers typically use either 24-bit or 32-bit color
 - with 24-bit color we have 3 bytes (remember 8-bits per byte)
 - one byte corresponds to the intensity of red to display, one byte corresponds to the intensity of green, and the final byte corresponds to the intensity of blue (remember RGB – red, green, and blue)
 - with one byte for each color, we can set the color intensity between 0 and 255 as we can store $2^8 = 256$ combinations in a byte.
 - If the byte is set to 0, the light bar for the color is turned completely off, if the byte is set to 255, the light is at maximum intensity.

¹ The term DPI is also sometimes used with computer monitors in place of PPI. However, the general trend seems to be using the term PPI for monitors and DPI for printers. You can think of the two as largely synonymous.


Here are a few sample color mixes:

| | | |
|---------|--|--|
| Red | RGB(255, 0, 0) |  |
| | in other words, Red at 255, Green at 0 and Blue at 0 | |
| Blue | RGB(0, 0, 255) |  |
| Purple | RGB(128,0,128) |  |
| Fuchsia | RGB(255,0,255) |  |
| Aqua | RGB(0,128,128) |  |
| Teal | RGB(0,255,255) |  |

You can create all sorts of interesting colors. For example:

| | | |
|------------|------------------|---|
| Sea Green | RGB(46,139,87) |  |
| Burly Wood | RGB(222,184,135) |  |
| Tomato | RGB(255,99,71) |  |

(The names for these colors are from the official HTML5 specification, which we'll talk about later in the quarter.)

The official specification for Stanford Cardinal Red is RGB(140,21,21) 

24-bit Color allows us to create ~16.7 million colors (actually 2^{24} colors). 24-bit color is sometimes referred to as TrueColor

- 32-bit color is the same as 24-bit color, except it adds the ability to control how transparent an object is.
 - We refer to the opacity of the object as **alpha**.
 - The last 8-bits in 32-bit color can control the transparency of the object allowing us to set its opacity to values between 0 and 255.
 - With alpha set to 0, the item is completely invisible.
 - With alpha set to 128, we can see both the item and the item underneath it.
 - With alpha set to 255, we can't see the item underneath it at all.
- HDR or High Dynamic Range²
 - Our ability to create better displays has increased considerably since 24-bit color was adapted.
 - New displays improve in several key areas:
 - We can create displays with higher pixels-per-inch (PPI) giving us more finely grained pixels.
 - The contrast ratio between the darkest black and the brightest white has increased.
 - The overall luminance or light output has increased.
 - Technologies such as 4K televisions and Apple's Retina Displays take advantage of our ability to create more finely grained pixels.
 - What about taking advantage of these other two capabilities?

² The term HDR or High-Dynamic Range is also used in photography. It is similar to HDR televisions and displays in that it is attempting to increase the range of color and light that can be represented. However, the technique is not the same (and in fact HDR photography precedes the digital era).

The advent of **HDR** or **High-Dynamic Range** is designed to take advantage of higher contrast ratios and higher overall luminance.

- There are several different competing HDR standards. However, all these standards increase the number of bits used to represent Red, Green, and Blue.
- The most common standard (HDR10) uses 10-bits
 - With 24-bit color we were able to set intensities of red, green, and blue from 0 to 255. Using the HDR10 standard we can set intensities of red, green, and blue between 0 and 1023.
 - Instead of 16.7 million colors, we have over a billion possible colors.
- A competing standard uses 12 bits each for Red, Green, and Blue giving us an even broader range.

How a Printer Displays Color

- In contrast to Display Technology's use of Red, Green, and Blue, most printers use a different set of primary colors: **Cyan, Yellow, Magenta, and Black**, typically abbreviated as **CMYK**.
 - The black isn't really there as a primary color, it can be created by mixing Cyan, Yellow, and Magenta, however, since it's the most common color used in printing computer documents, using a separate (and often larger) Black ink reservoir allows us to save on the other inks.
- We use different primary colors because Printed pages and Computer Displays create colors in different ways.
 - Computer Displays generate light. They shine varying amounts of Red, Green, and Blue directly into your eyes.
 - This is called **Additive Color**, since they add Red, Green, and Blue together.
 - Printing and Painting don't generate light directly. Instead, on overhead light or the sun shines down on the painting. Ideally this light is pure white light and contains all the parts of the color spectrum. The ink on the printed page absorbs some parts of the light spectrum, and the remaining parts of the color spectrum are reflected into your eyes.
 - This is called **Subtractive Color**. If I mix both magenta and yellow ink together then my resulting painting will absorb both the parts of the light spectrum that magenta would have absorbed and the parts of the light spectrum that yellow would have absorbed. Only the parts absorbed by neither paint will be reflected on to me.
 - When shopping for Color Wheels (a common tool used in both painting and in web development where it can help choose website colors) you'll discover different types of Color Wheels
 - Some Color Wheels are designed for Subtractive Color, while others are designed for Additive Color

Storing Images

- Now that we've seen how the displays themselves work, we next take a look at how images are stored on the computer.
- There are two fundamentally different approaches we can take in representing an image on the computer

- Using a **Bitmap** approach (also known as Raster graphics), we think of an image as being represented by all the individual pixels in the image and the color of those individual pixels.
- Using an **Object** approach (also known as Vector graphics), we think of an image as being composed of a number of different geometric shapes.
 - Using this approach, I would think of an image as consisting of lines, ovals, rectangles, etc. I would store the image by storing those geometric shapes along with their dimensions, colors, and line widths.
- Each of these approaches has their advantages and disadvantages

Object Advantages

- Storing images as objects takes less space (since I just need to remember a few numbers for each shape, such as its coordinates and color, rather than a whole bunch of individual pixel values).
- Remembering images as geometric shapes allows me to zoom in and zoom out and still have the image look crisp, because I'm using mathematical formulas to determine what to draw on screen. In contrast zooming in on a bitmap displays large blocky pixels.
- Objects are easy to manipulate, I can draw a circle, and the computer still remembers it's a circle, so I can grab that circle, move it, and resize it.

Bitmap Advantages

- The main advantage of bitmaps is that some kinds of images simply can't be represented using the Object approach. A complex photograph cannot be realistically broken down into a whole bunch of geometric shapes.

We're going to primarily focus on Bitmap Graphics from here.

- most images used on the web are stored as bitmaps
 - That's partly because historically there has not been an object-graphics standard for the web, now however SVG (Scalar Vector Graphics) is widely supported – we may look at it later in the quarter.
- digital photographs are also based on bitmap graphics, object graphics use is generally more specialized
- just as with display screens, bitmaps have their own height and width in pixels and their own color depth
 - the right number of pixels to use for a bitmap depends on its intended purpose
 - placing a bitmap with a high number of pixels on a webpage will require a lot of time to download the image from the web server and will cause the webpage to display very slowly
 - trying to print a bitmap with a small number of pixels on a printer will either make for a very tiny image or a very blurry image.
 - printers have much higher dots-per-inch (DPI) than a monitor has pixels-per-inch (PPI).
 - a very high-quality display might have around 250 dpi, in contrast a high-end printer could be rated at 600dpi, 1200 dpi, or even higher.

- as a result, high numbers of pixels in an image are very useful when printing
- don't take your high pixel-count images and reduce them in size if you intend to print the image at a later time

How Storage of Images Work

- The simplest approach to storing an image is to simply remember all the pixels
- However, this isn't the most efficient approach.
- Is it possible to remember all the information in an image, but do it using a method other than simply storing all the values of each individual pixel?
- Suppose I have a 1024x768 pixel image and that image is primarily showing just a red background, but in the top-left corner I have a capital "H" displayed in black.
 - I am talking on the telephone to someone else, trying to accurately convey this image to them.
 - Using the standard approach, I would simply read the RGB value of every single pixel in the image.
 - Pixel (0,0) has RGB(255,0,0), Pixel (1,0) has RGB(255,0,0), Pixel (2,0) has RGB(255,0,0) ...
 - Clearly, this would accurately transmit the image over the phone.
 - It would also be very time consuming and very tedious.
 - As an alternative, I could simply say, "the entire first line is all RGB(255,0,0), the second line is the same as the first line, the third is the same as the first line." I would have to specifically describe the section of the image with the "H" on it, but even then, with a capital "H" the sections of the "H" above the crossbar would be identical to the sections of the "H" below the crossbar.
 - I've changed how I'm describing the image, but someone could still accurately reproduce the image from my description.
 - This technique of describing an image by describing which sections are the same as other sections is actually one of several techniques used to reduce the size of image files.

Compression and Decompression

- Internally when an image is being displayed on a computer screen, all computers maintain a grid in computer memory storing the color values to display for each pixel. I will refer to this as "bitmap format".
- However, this direct, straightforward method isn't used for file storage on the web, because it takes a lot of space, and for webpages, a file format that takes a lot of space. As I mentioned briefly above, this will result in the image taking a long time to download and will result in the webpage taking a long time to display.
- Instead we use other file formats that take up less space. We've just seen in the "H" example above, how we can represent an image in different ways, and how some of those ways of representing the image might take less space than others.
- When we convert from the original bitmap format to another format which takes less space we say that we have **compressed** the image. When we convert from that

compressed file format back to the original bitmap format we say we are **decompressing**.

There are three different image file formats widely used on the web – JPEG, PNG, and GIF.³ Let's take a close look at what their advantages and disadvantages are.

JPEG

- **JPEG is designed to store photographs**
- It allows the use of 24-bit color
- It is lossy, which means that if we take an image and store it in JPEG, we will not be able to convert it precisely back to the original. Some information will be lost in the transformation.
- JPEG introduces what are called **compression artifacts**. These are imperfections or errors which are introduced into the image as a result of converting to the JPEG format.
- Here is an example—the image on the left is the original, the image on the right is displaying JPEG compression artifacts:



- If you look very carefully at the railing running along the roof of the yellow building, you'll see that the railing is not as smooth:



³ You may also run into the WebM format, which is sponsored by Google. I do not recommend using this format, because as of this writing, Apple does not support it and shows no signs of planning to add it to any of their web browsers.

- You'll see something similar if you look at the wires holding up the façade on the left:



- In fact, if you look closely you'll often see artifacts anytime a dark color section and a light color section meet:



- If JPEG introduces these compression artifacts, why do we use it?
 - Because it makes the file much smaller, and the artifacts are often not very obvious anyway. Would you have noticed these differences if I hadn't pointed them out?
 - JPEG allows us to control how much compression takes place. Generally, compression artifacts won't be too obvious unless you really crank up the compression rates – this leads to a much smaller file, but potentially some very obvious compression artifacts.
 - The two images I show here are both excerpts from a much larger photo. The original photo is 1.39 Megabytes, whereas the compressed version is 215 kbytes. ***The original is over 6.5 times larger. So I saved a huge amount of space with the introduction of some very subtle issues.***
- JPEG is very space efficient, and unless the compression rate is turned up high compression artifacts aren't particularly noticeable, because of this virtually ***all mid-to-low-end consumer cameras including phones just store directly to JPEG.***

High-end cameras often have a setting to store to RAW format instead of to JPEG. This means that these cameras can store the grid's original pixel values, with no JPEG compression. There is a tradeoff though, as these will take a great deal more storage space.

- If you're curious, JPEG stands for Joint Photographic Experts Group – the name of the committee that came up with the standard.

PNG

- PNG allows for the use of 24-bit color or 32-bit color including alpha transparency.
- In contrast to JPEG, ***PNG compression is lossless.*** This means that our PNG copy is precisely the same as our original.

- PNG uses a technique very similar to the technique I described with the red background with an “H” on it. It looks for repeated sections of an image and rather than simply repeating all the individual pixel values, it simply refers to the previous section.
- PNG does not work well with photographs – photographs don’t really have sections that are exactly duplicates of other sections.
 - You can store a photograph with PNG, but you won’t get very good reduction in file size.
- **PNG works well for logos, diagrams,** and other types of images where sections of an image are repeated.
 - In this Stanford logo, many sections of the logo are exactly the same. The first few lines are exact duplicates of each other (just pure red) as are the last few lines. While the middle lines aren’t exact duplicates, large portions of them are the same.
- PNG stands for Portable Network Graphics (but that’s probably not worth remembering)



GIF

- Our last format to discuss is GIF.
- This is an older format.
 - PNG was specifically designed to replace GIF.
- GIF supports only 256 colors.
- It supports the ability to make one of those colors transparent.
- At this point GIF is used for one purpose, creating short animated loops, where a sequence of images plays over and over again.
 - The newer PNG format does have an animated version APNG (Animated PNG), however, as of this writing it’s only supported by 75% of the browsers in use worldwide (it’s still not supported by Microsoft Internet Explorer or their newer Edge browser nor is it supported by the Samsung Browser used on their phones).
- I definitely don’t recommend using GIF unless you need it for short animated loops

Summary of Bitmap Web Storage Formats

- Use JPEG when storing photographs
- Use PNG when storing graphics like logos or diagrams.
 - You may also use PNG if you’ve got a photo but need to take advantage of PNG’s transparency effects, just be aware that the file will be much larger than the equivalent JPEG.
- Only use GIF when you’ve got a short animated loop.

Summary of Compression

- Compression is a means of taking a large amount of data and storing it in a smaller amount of space.
- Compression can be lossless, which means no information is lost during compression.
- Compression can also be lossy, which means we’ve lost some information in our attempt to make the data fit in a smaller amount of space.
- Compression isn’t just for images
 - The widely used ZIP format compresses files.

- You may have used ZIP to compress files for submission for a class.
 - Is this lossy or lossless compression?⁴
 - MP3, Apple AAC, and Windows WMA music formats all use compression. We'll take a look at how they work next lecture.
- Even lossless compression isn't free. Compressing data takes CPU resources. Decompressing data also takes CPU resources.

⁴ If you think about it, using lossy compression to store Word or Text documents or CS106A Java Projects to submit for class is a bad idea. You want your instructor to see the work you actually did, not something with compression artifacts. ZIP is a lossless format.