# IP Workshop: Image Basics

Mark M. Meysenburg

Doane University

*mark.meysenburg@doane.edu*

May 10, 2017

# Overview

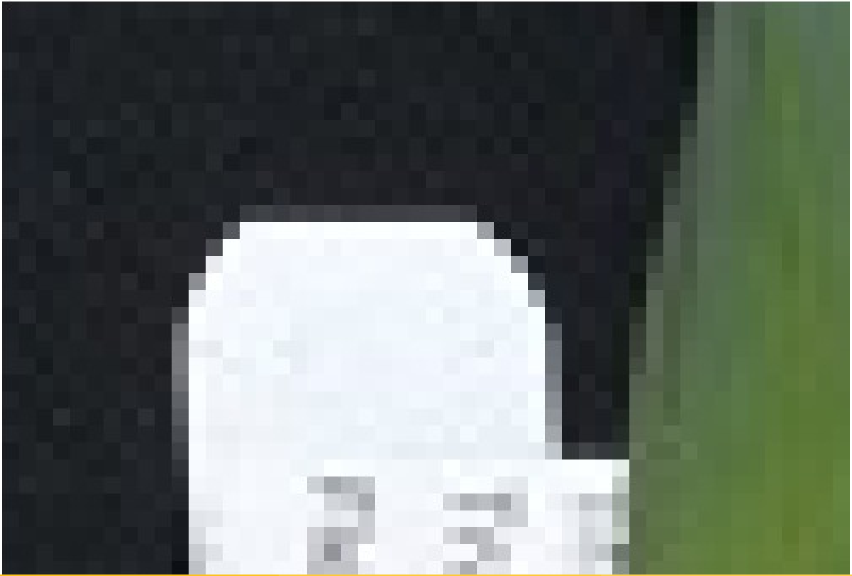
1. Pixels
2. Coordinate system
3. Color model
4. Image formats
   - BMP images
   - Image compression
   - JPEG images
   - TIFF images
5. Image metadata

# Introduction to pixels

- Digital images are stored as rectangular arrays of hundreds, thousands, or millions of discrete "picture elements," otherwise known as pixels. Each pixel can be thought of as a single square point of colored light
- Each is all one color, but each pixel can have a different color from its neighbors. Viewed from a distance, these pixels seem to blend together to form the image we see

# Looking at pixels

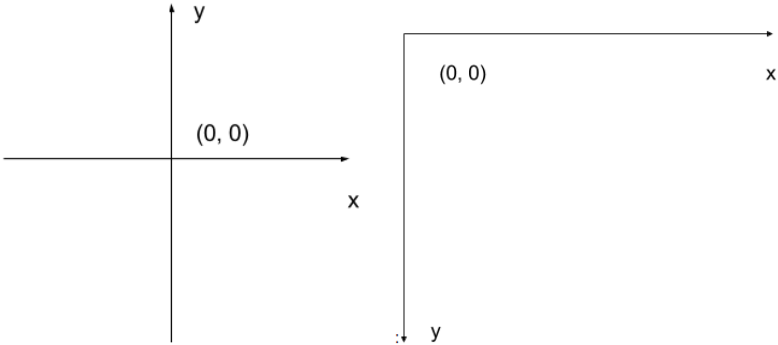
# Looking at pixels



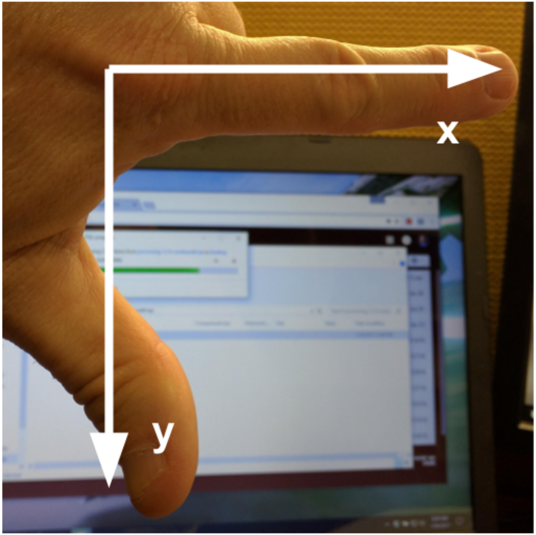
# Image coordinate system

- We can access, examine, and / or change the color of any pixel we wish. To do this, we need some convention on how to access pixels individually; a way to give each one a name or an address of sort
- We use the *left-hand* coordinate system to specify the location of individual pixels

# Image coordinate system



# Image coordinate system



# Introduction to color models

- Digital images require some *color model* to create a broad range of colors from a small set of primary colors
- There are several different color models that are used for images, the most commonly occurring one is the additive red-green-blue (*RGB*) model
- In the RGB model, different intensities of red, green, and blue are combined to create the color for the image pixel

# Details of the RGB model

- Each of the primary colors in the RGB model is often called a *channel*
- Usually, the amount of the primary color added is represented as an integer in the closed range [0, 255], giving 256 discrete values for that channel
  - 256 comes from the number of bits used to represent the value
  - Eight bits per channel gives the common *24-bit color*
- Any color in the RGB model can be expressed by a triplet of integers in [0, 255], representing the red, green, and blue channels, respectively. A larger number in a channel means that more of that primary color is present
- See, for example, `http://www.rapidtables.com/web/color/RGB_Color.htm`

# Details of the RGB model

- With 24-bit color, we can represent $2^{24} = 16,777,216$ distinct colors
    - The human eye can only distinguish approximately $10,000,000$ colors, so that seems to be plenty...
    - ... unless you are doing science that looks at colors humans cannot distinguish!
- Other common color depths:
    - 8-bit color: 3 bits red, 3 bits green, 2 bits blue, for $8 \times 8 \times 4 = 256$ colors
    - 16-bit color: 4 bits for each (plus 4 for *alpha*), giving $16 \times 16 \times 16 = 4,096$ colors
    - 18-bit color: 6 bits for each, giving $64 \times 64 \times 64 = 262,144$ colors

# Image formats

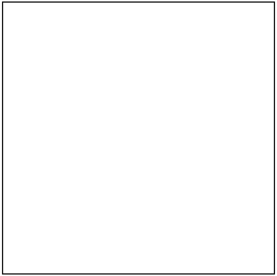
- Abstractly, we think of digital images as instances of *raster graphics*
  - Rectangular arrays of pixels, with the color for each pixel specified by a RGB triplet
- But, images are not necessarily stored in the computer, on disk, or transmitted over a network in that format
- There are several digital image file formats we might encounter:
  - Device-Independent Bitmap (BMP), with `.bmp` extension
  - Joint Photographic Experts Group (JPEG), with `.jpg` or `.jpeg` extension
  - Tagged Image File Format (TIFF), with `.tif` extension

# BMP images

- Closest file format to our raster graphics conceptualization
- Supports 8-, 16-, or 24-bit color
- Advantages:
  - Simple file format
  - High quality images
  - Viewable by almost every system / application
- Disadvantages:
  - Images are not compressed, so file sizes are very large

# Image compression

Thought experiment: Imagine a $5,000 \times 5,000$ pixel image, composed of nothing but white pixels. How much storage space would be required to store the image?



# Image compression

- Since image files can be very large, various compression schemes exist for saving (perhaps approximately) the same information while using less space
- These compression techniques can be categorized as *lossless* or *lossy*

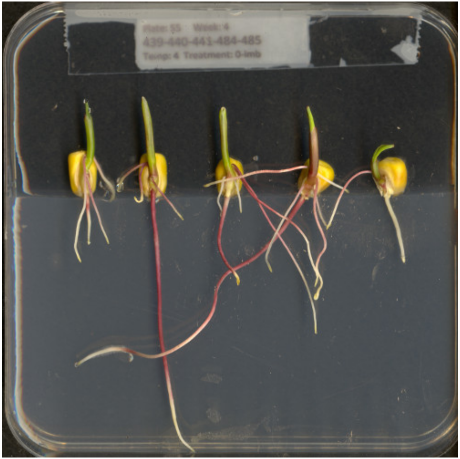
# Lossless compression

- In lossless image compression, the computer applies some algorithm to the image, resulting in a file that is significantly smaller than the uncompressed file equivalent would be. Then, when we wish to load and view or process the image, our program reads the compressed file, and reverses the compression process, resulting in an image that is identical to the original. Nothing is lost in the process hence the term "lossless."
- See the white-square-making program

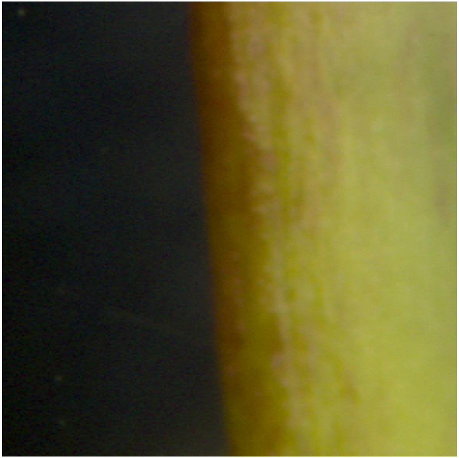
# Lossy compression

- Lossy compression takes the original image and discards some of the detail in it, resulting in a smaller file format. The goal is to only throw away detail that someone viewing the image would not notice. Many lossy compression schemes have adjustable levels of compression, so that the image creator can choose the amount of detail that is lost. The more detail that is sacrificed, the smaller the image files will be – but of course, the detail and richness of the image will be lower as well.
- It is important to understand that once an image is saved in a lossy compression format, the lost detail is just that – lost. I.e., unlike lossless formats, given an image saved in a lossy format, there is no way to reconstruct the original image in a byte-by-byte manner.

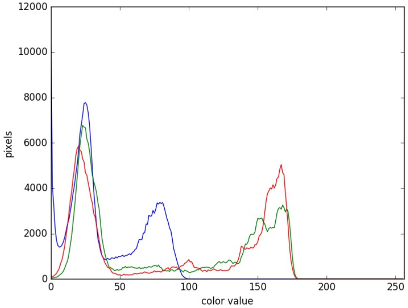
# Lossy compression example



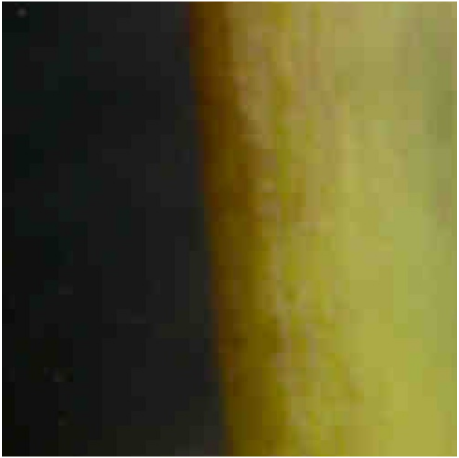
# Lossy compression example



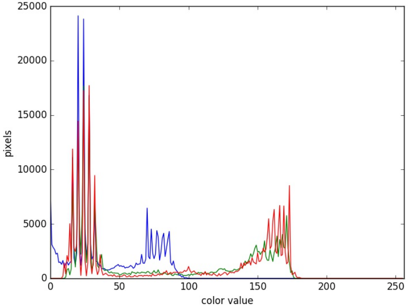
# Lossy compression example



# Lossy compression example



# Lossy compression example



# JPEG images

- Most common digital image format
- Uses tunable, lossy compression
- 24-bit color depth
- Advantages:
  - Viewable by almost every system / application
- Disadvantages:
  - Lossy compression may remove needed detail from images
- Rule of thumb: If you are using JPEG images, set the compression rate low to preserve as much detail as possible

# TIFF images

- Popular with publishers, graphics designers, and photographers
- Can be uncompressed, or compressed using either lossless or lossy compression schemes
- Advantages:
  - High quality with small file size (if lossless compression is selected)
- Disadvantages:
  - High file size or loss of detail, depending on compression options
  - Not universally supported in all systems / applications

# Image metadata

- *Image metadata* is textual information that is contained within an image file, but not normally shown when the image is viewed
- Metadata holds information about the image itself, such as when the image was captured, where it was captured, what type of camera was used and with what settings, etc.
- Users can also edit metadata, adding custom information to images
- JPEG and TIFF images support the inclusion of metadata in images

# Metadata example

Using ImageJ, examine the metadata for this image.