Lecture 6:

Importing and displaying images

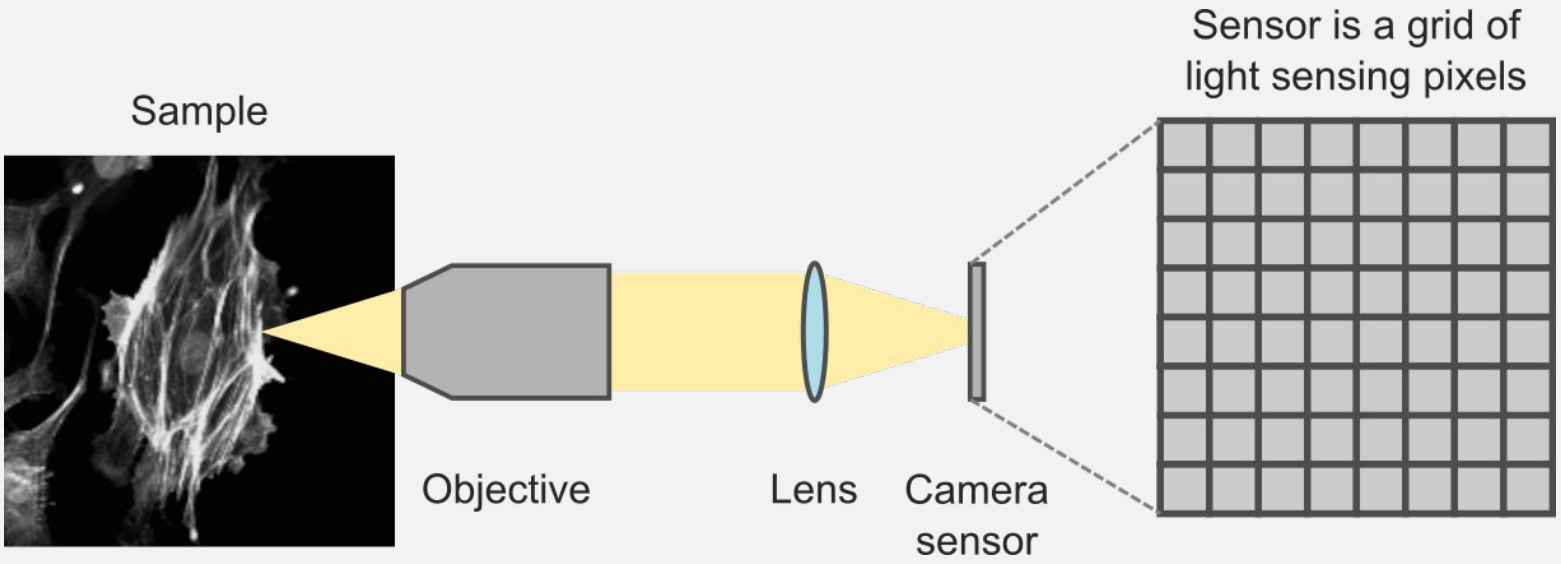
Lecturer: Jian Wei Tay

Date: 3 September 2021

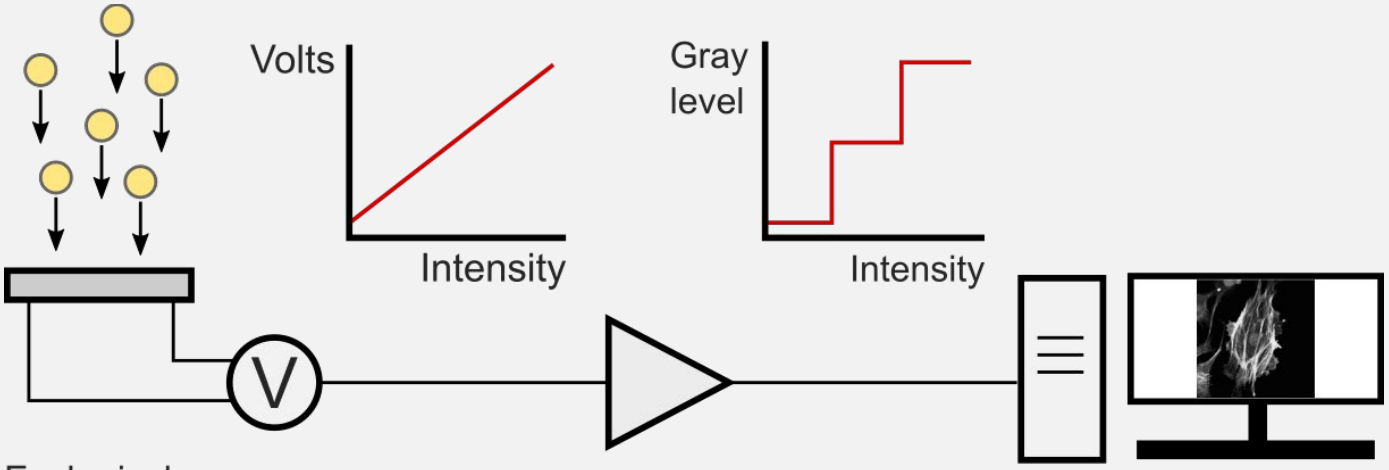
Learning objectives

- How digital images are recorded and stored
- Reading and displaying images into MATLAB
- Integer and double data types
- Making simple image measurements

How are digital images made?



What happens on a single camera pixel?

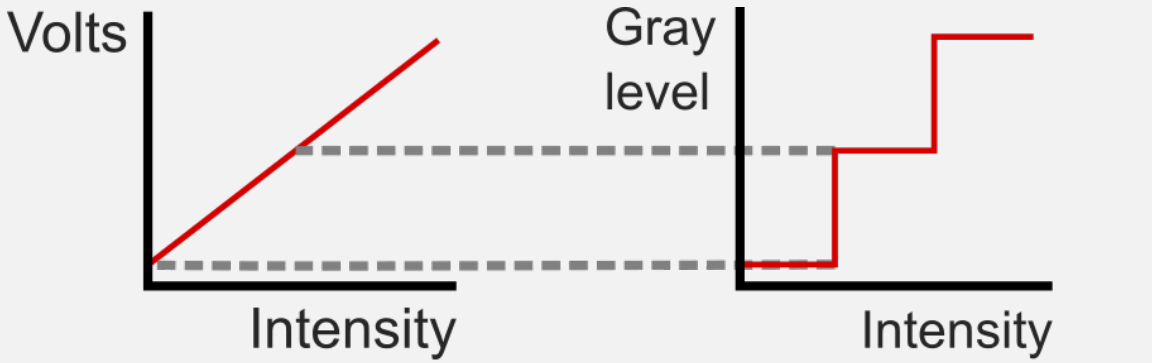


Each pixel measures the amount of light (intensity) that falls on it during the exposure period

The analog voltage is digitized into discrete gray levels

Which is then read by the computer

Digitization means binning intensity



Each gray level represents a range of intensities

Digital images are made up of pixels

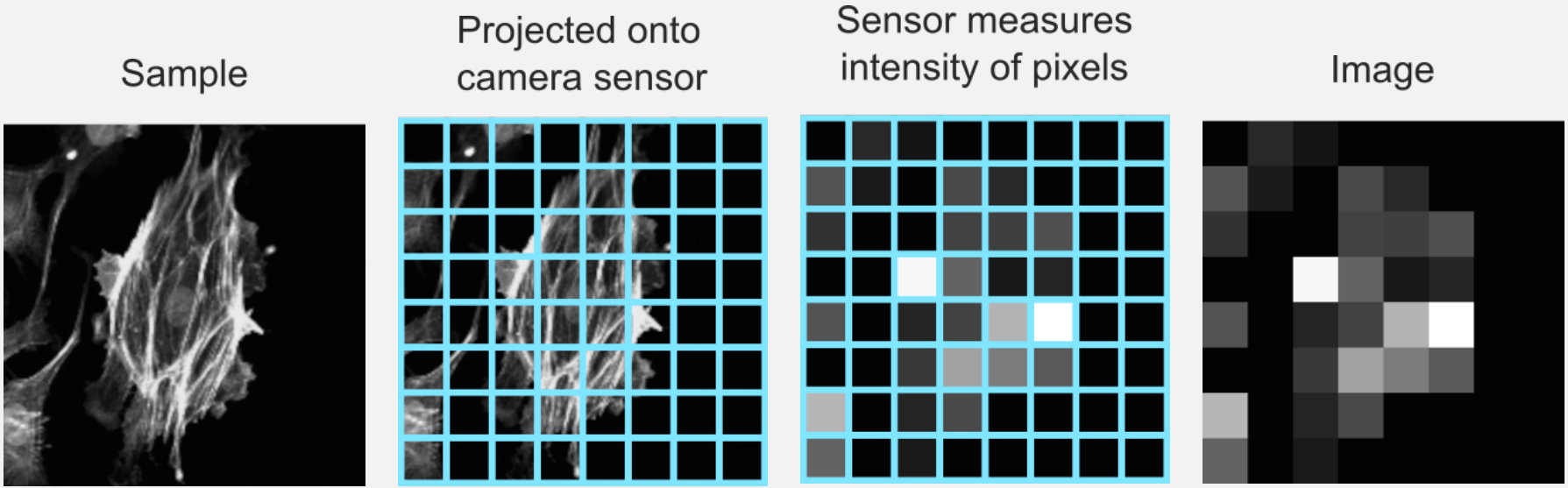
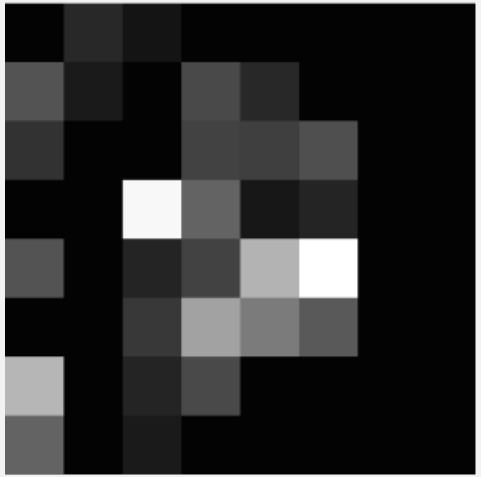


Image pixel size = Area of original sample projected on to a single camera pixel (and therefore forming a single image pixel)

Image data is stored as an array of numbers

Image



Data

3	40	20	3	3	3	3	3
83	26	3	70	40	3	3	3
50	3	3	66	63	79	3	3
3	3	248	99	23	36	3	3
83	3	36	66	179	255	3	3
3	3	56	162	123	89	3	3
182	3	36	73	3	3	3	3
99	3	26	3	3	3	3	3

Note: Each matrix element corresponds to one pixel of the image.

Questions?

Reading an image

```
I = imread('filename')
```

- The function `imread` will read basic image file types (e.g. .tiff, .jpg, gif), returning the data in matrix `I`

Note: The file needs to be either in the current folder or on the MATLAB path. Otherwise, specify the full path to the image, e.g., `'C:\Downloads\image.tif'` on Windows.

Also note that the single quotes `''` are needed to specify text in MATLAB.

Practice

- Read in and display the image `l6_cardiomyocytes.tif` into MATLAB

Question

- What is the size of the image in pixels?

A. 1280 x 900

B. 601 x 901

C. 901 x 601

D. How do you get the size of an image?

Displaying an image

```
imshow(I)
```

- The function `imshow` will display the data in matrix `I` as an image

Question

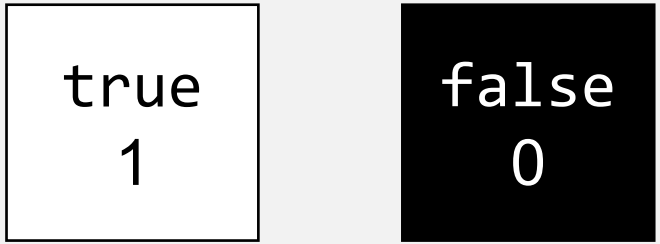
- What is the data type of the image data?

- MATLAB lists the data type next to the size in the Workspace

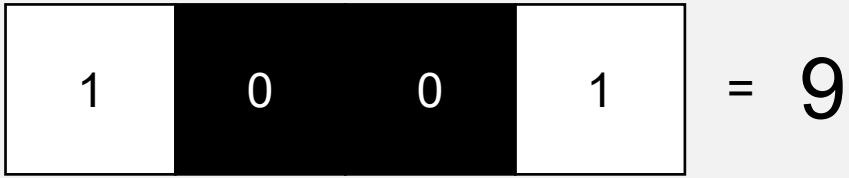
Digital data is stored as bits

- Data is stored electronically as bits

- A single bit can only have one of two values:



Larger values are stored by grouping bits together



Data types

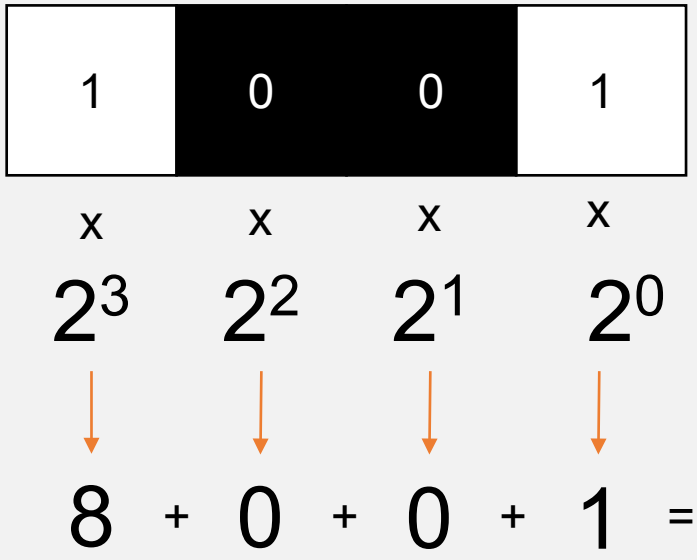
The data type of a variable tells the computer how to interpret the bits

Important data types to know

- Image data is typically stored as unsigned integers

- Numbers in MATLAB are typically double-precision floating-point numbers (or doubles for short)

Example unsigned integer



4-bit unsigned integer

The position of the bit indicates its value as a power of 2

Unsigned integers

- Can only represent whole numbers (no decimal places)
- Range of values = 0 to $2^N - 1$
- Typically specified in multiples of 8-bits (e.g. 8-bit unsigned integer = uint8, 16-bit unsigned integers = uint16)

Note: Unsigned integers can only hold positive numbers – perfect for images!

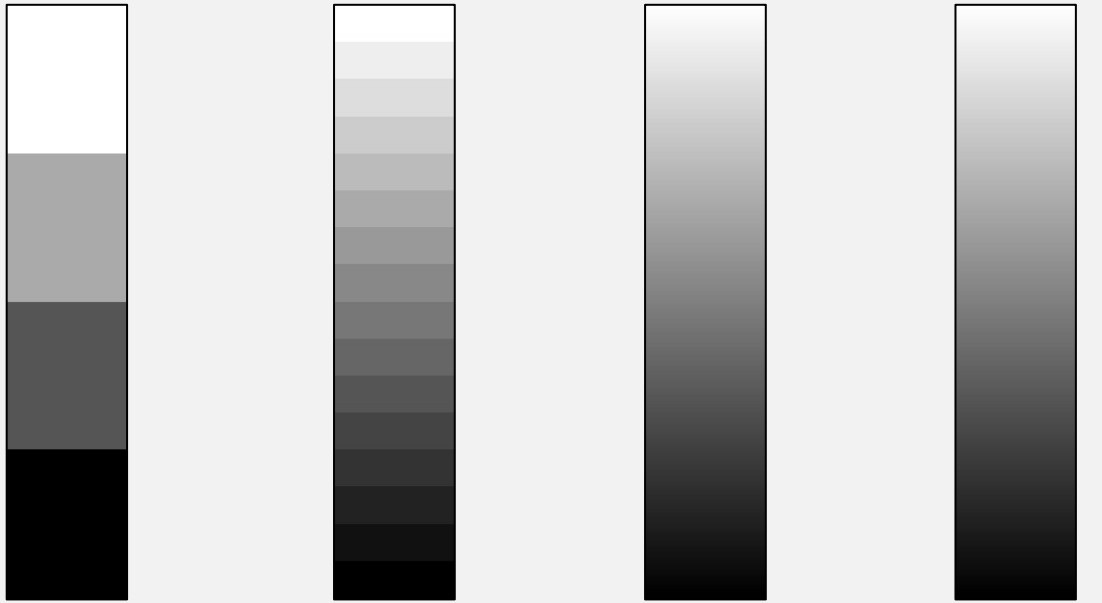
Question

- What is the data type of the image we read in earlier?

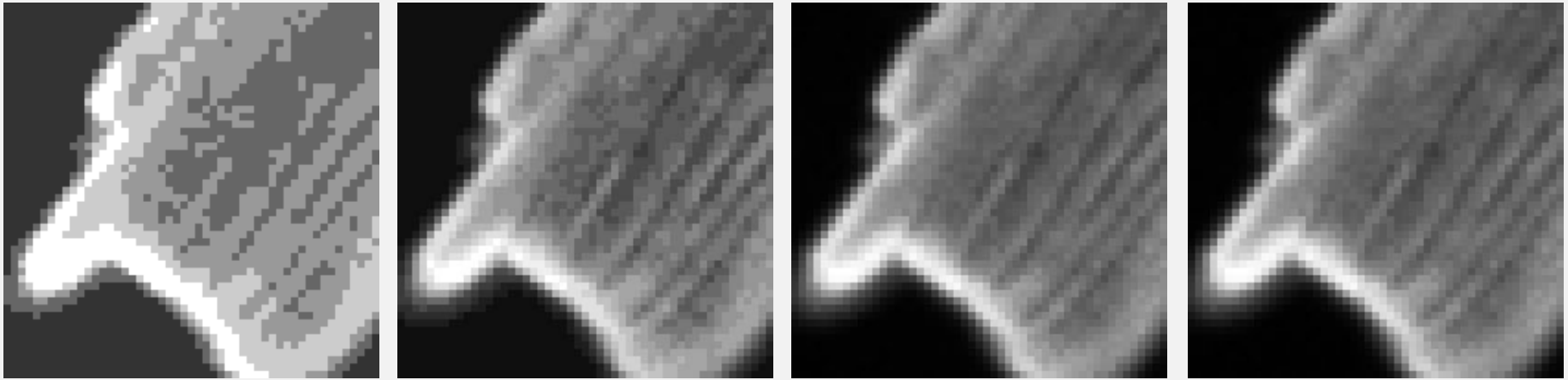
Question

- What is the range of this data type?

How the number of bits affect image quality



How the number of bits affect image quality



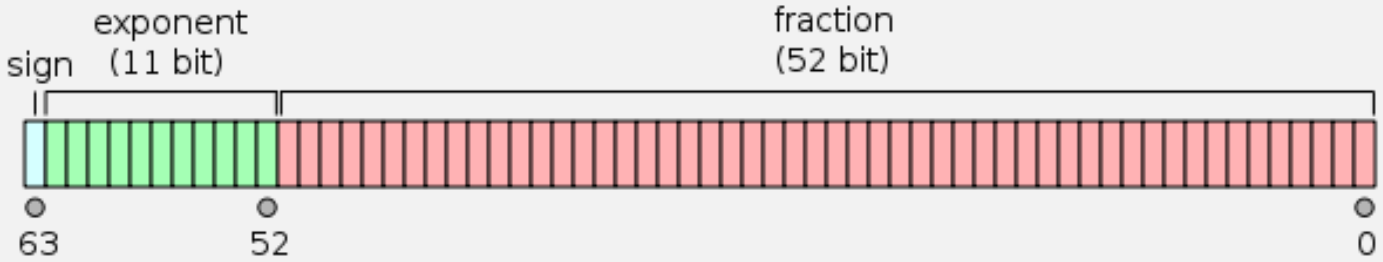
2 bit 4 bit 8 bit 16 bit

More accurate representation →

Double-precision floating-point number formats

- Supports positive and negative numbers
- Supports decimal places
- Is 64-bits long
- Has a more complicated layout:

Note: You don't need to know exactly how doubles work, but you need to know why and when to convert unsigned integer image data into double format.



Questions?

Measuring object length

- The function `imdistline` will create a line on the current figure

- Display an image first using `imshow`, then use `imdistline`

Units

- `imdistline` provides measurements in pixels

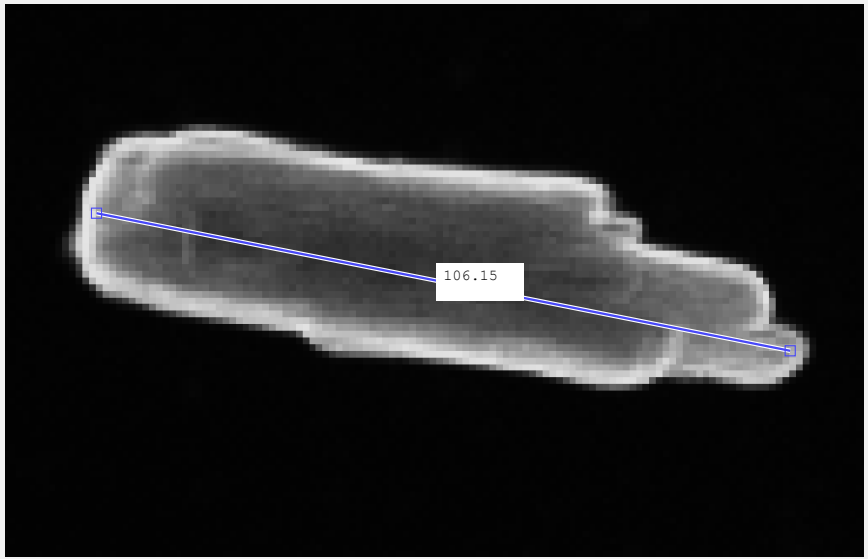
- To convert, multiply by the image pixel size

Size in μm = Size in pixels x Image pixel size

$$\cancel{px} \times \frac{\mu m}{\cancel{px}}$$

Question

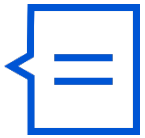
- If the image pixel size is 0.9174 μm/px, what is the actual size of the cell shown below (106.15 px)?



Reading of pixel intensities

- Using the data tip tool

Note: Hover over the image to get the toolbar to appear. The data tip icon looks like:



- By indexing from the image matrix

Question

- What is the command that indexes the pixel shown below?

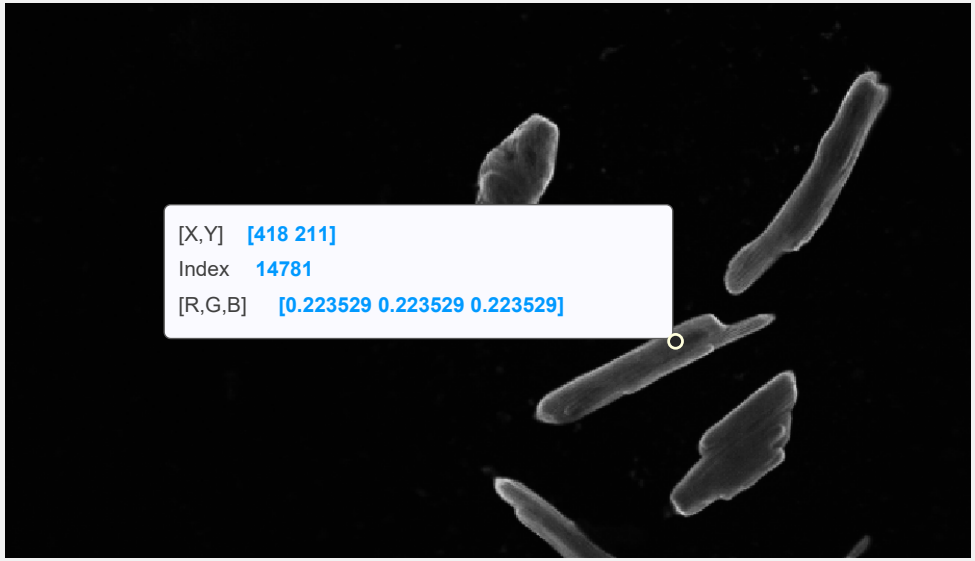
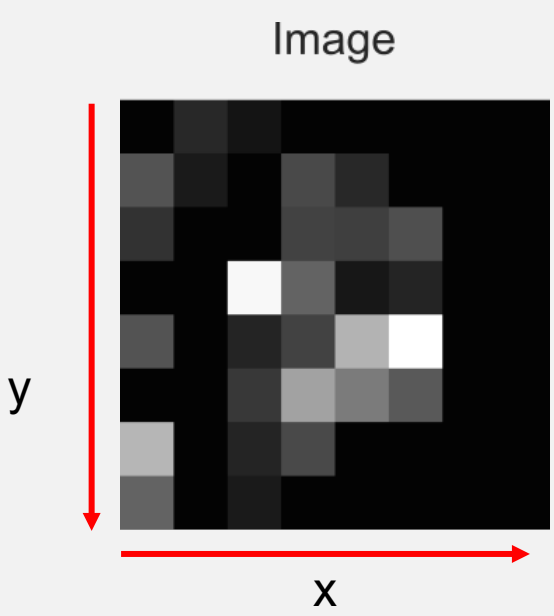
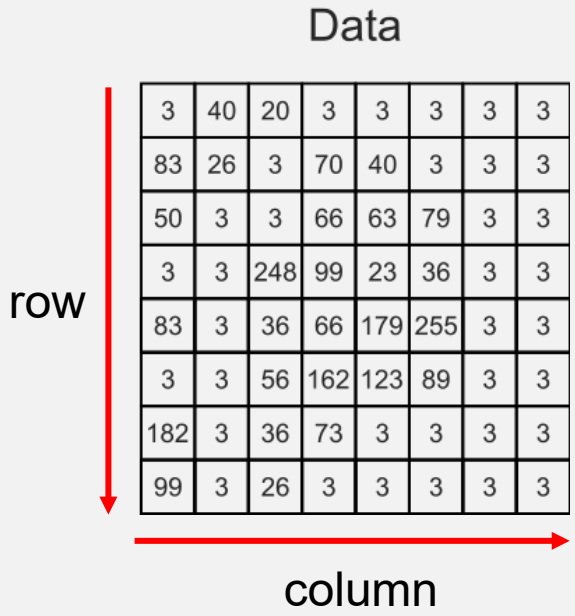


Image vs matrix coordinates





3	40	20	3	3	3	3	3
83	26	3	70	40	3	3	3
50	3	3	66	63	79	3	3
3	3	248	99	23	36	3	3
83	3	36	66	179	255	3	3
3	3	56	162	123	89	3	3
182	3	36	73	3	3	3	3
99	3	26	3	3	3	3	3

Note: Remember to "swap" coordinates when indexing, i.e., `I(y, x)`

Unit of pixel intensities

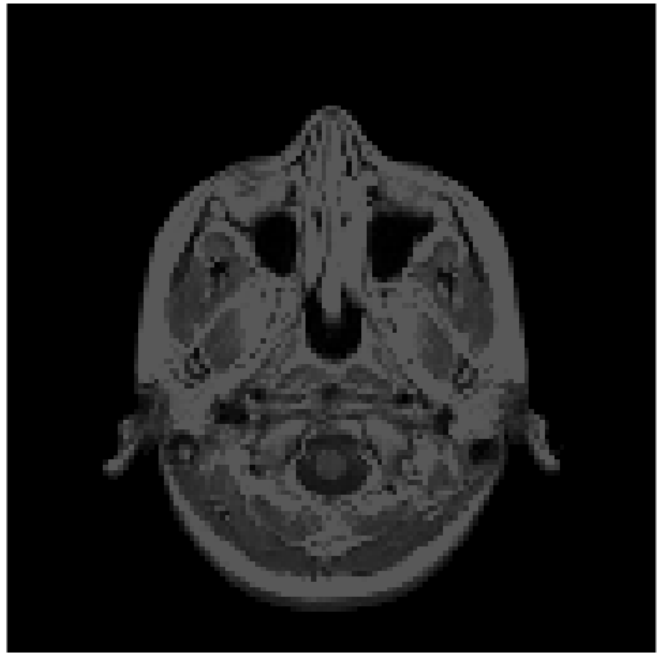
- Actual conversion needs proper calibration of microscope and camera – most people don’t do this
- Usually reported in “counts” or “arb. units”
- Most measurements are reported as unitless relative values (e.g., twice as bright)

Questions?

Image processing – Adjusting contrast

- Read in an example image:

```
>> I = imread('mri.tif');
>> imshow(I)
```

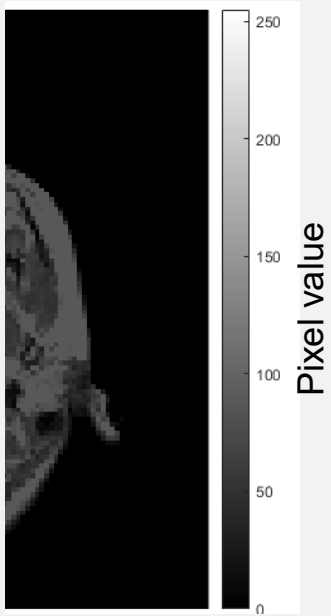


Note: This image looks dark to the eyes because there isn't a lot of contrast (i.e., there isn't much difference between the light and dark regions of the image)

Displaying the colorbar

The color bar displays which pixel values are mapped to which color

```
>> colorbar
```

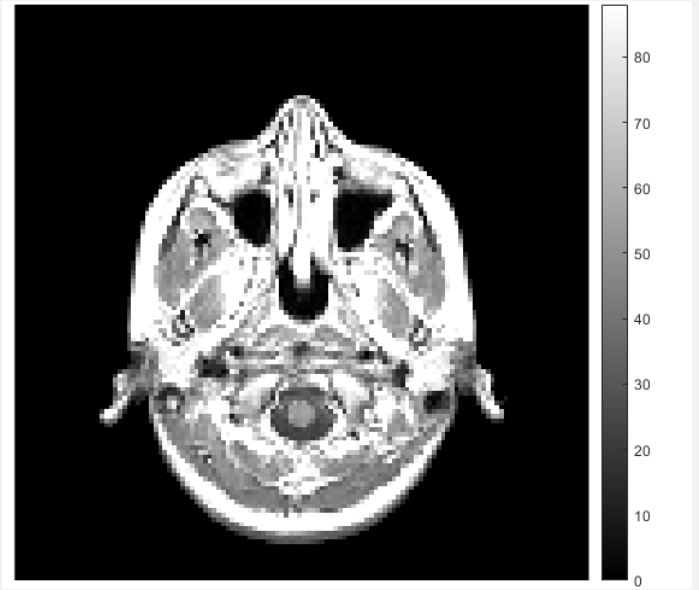


Note: By default, `imshow` scales the colorbar depending on data type. For a uint8 image, black = 0 and white = 255 (the maximum integer value of an 8-bit integer)

Automatic display scaling

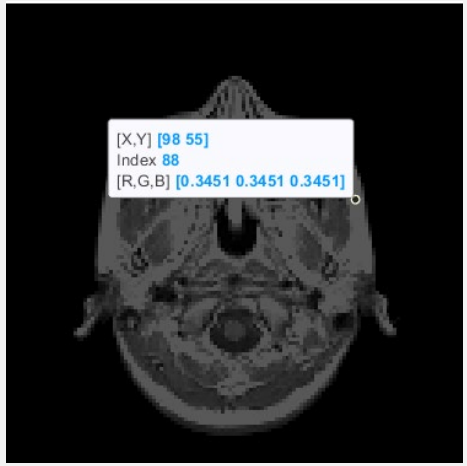
```
imshow(I, [])
```

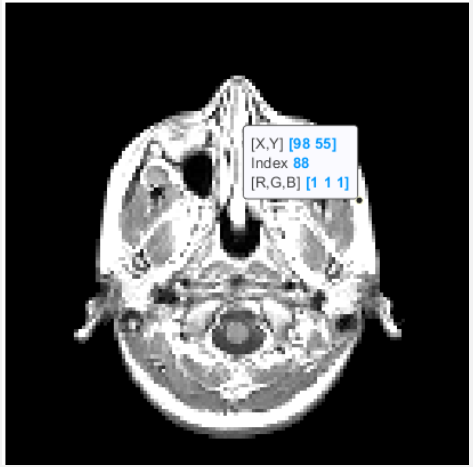
- Adding an empty matrix `[]` as the second input argument to imshow will scale the image so white = maximum pixel value and black = minimum pixel value



Changing the displayed color scale manually

```
imshow(image, [low, high])
>> imshow(I, [0, 88])
>> colorbar
```





Note: The underlying data (see the index value) was not changed, just how the data was displayed.

Adding false color

- Color microscope images are actually false colored after the images are acquired

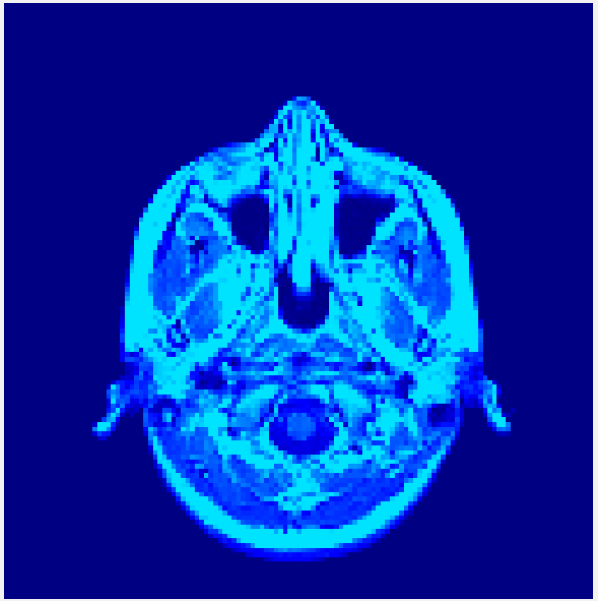
- In MATLAB, you can apply a color map to a greyscale image, e.g.:

```
>> colormap('hot')
```

Hint: You can see all color maps in the documentation by typing `doc colormap`

Example

```
>> I = imread('mri.tif’)
>> imshow(I)
>> colormap(‘jet')
```

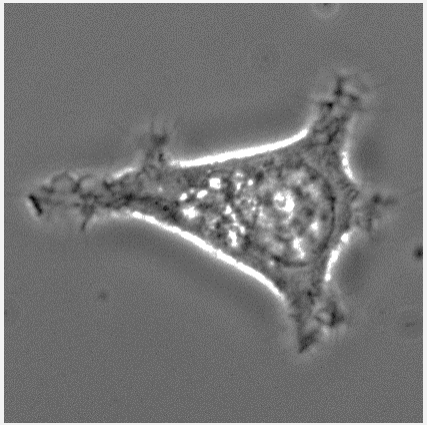


Saving images

- You will need to print some images to submit with your homework

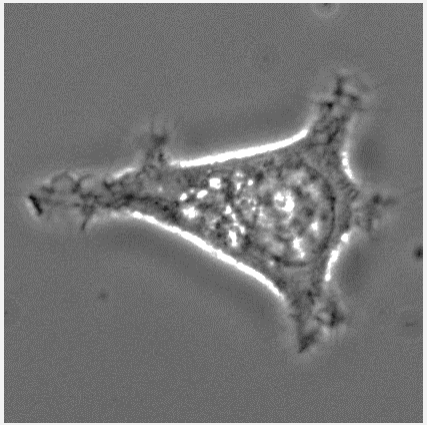
- You can save images using File > Save As in the Figure Window
  - You can save these as .jpeg or .gif formats

Never use JPEG or GIFs for analysis



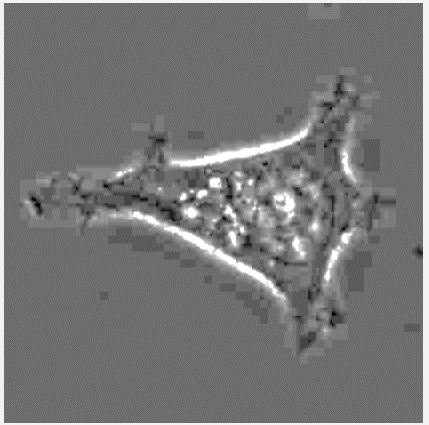
Original
Uncompressed TIFF

535 kB



40%
JPEG compression

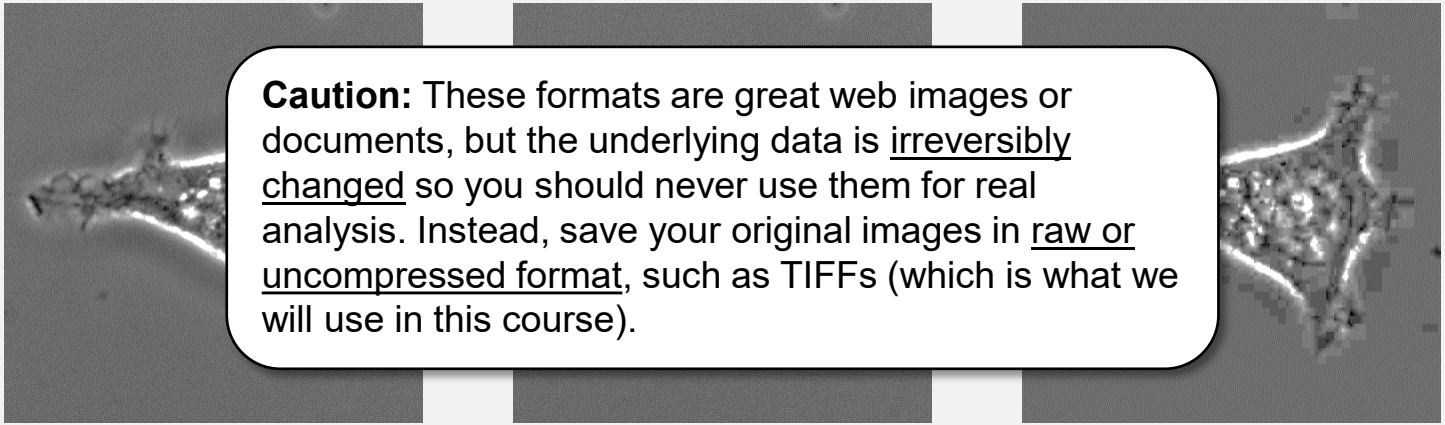
19.8 kB



95%
JPEG compression

10 kB

Never use JPEG or GIFs for analysis



Caution: These formats are great web images or documents, but the underlying data is irreversibly changed so you should never use them for real analysis. Instead, save your original images in raw or uncompressed format, such as TIFFs (which is what we will use in this course).

Original Uncompressed TIFF	40% JPEG compression	95% JPEG compression
535 kB	19.8 kB	10 kB