Light Microscopy Core Facility
DUKE UNIVERSITY AND DUKE UNIVERSITY MEDICAL CENTER

CMB 551 Module 1A

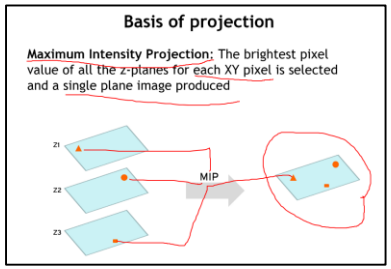
Image processing and quantitative image analysis

Sam Johnson

Benjamin Carlson

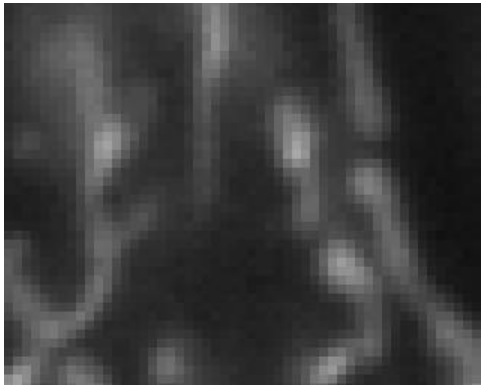
http://microscopy.duke.edu/ benjamin.carlson@duke.edu

n



```
File:
Filename_to_open
```

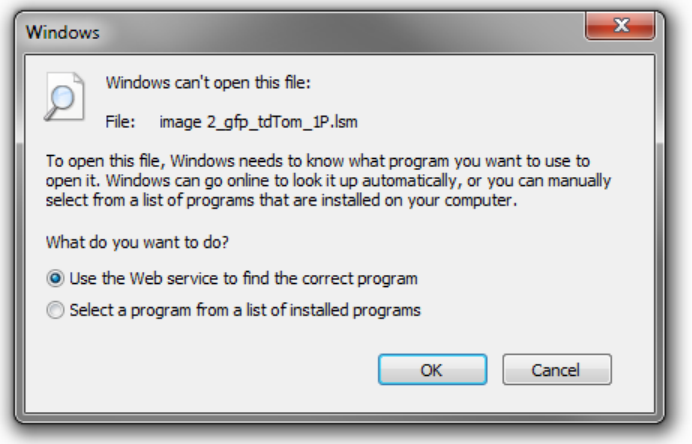
Microscopy images are somewhat different to other fields

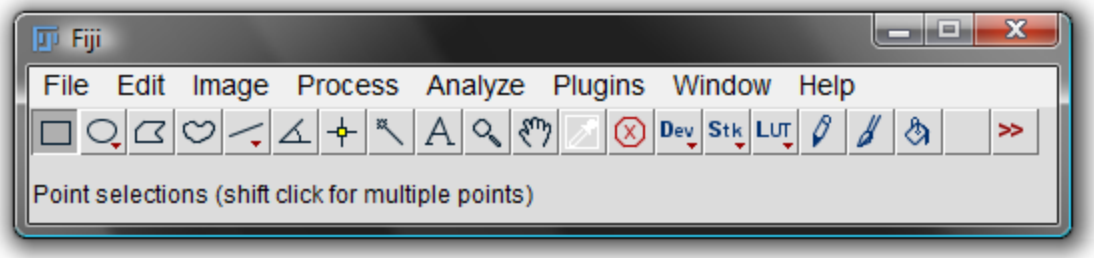


Microscopy data are in many formats

Microscope specific formats

LIF, LEI, LSM, ZVI, STK, OIB, OIF, dv, ICS/IDS, ICS2, r3d, PIC, IPL, CXD, SDT, IPW, ND2, RAW, IMS. . .





- ✓ Types of images – bitmap/vector and compression
- ✓ Bit-depth, histogram, scaling and gamma
- ✓ Export to standard formats for figures
- ✓ Color use and misuse
- ✓ 3D images – Projections and other views
- ✓ Stitching and alignment
- ✓ Image processing – Filters
- ✓ Segmentation – Regions of interest
- ✓ Quantification – Count, measure, intensity, colocalization, 3D measurements, tracking

Data → big(ish) data

One image	= ~1 MB
λ*3 Z*11 t*360	= ~12 GB

Can be a challenge to

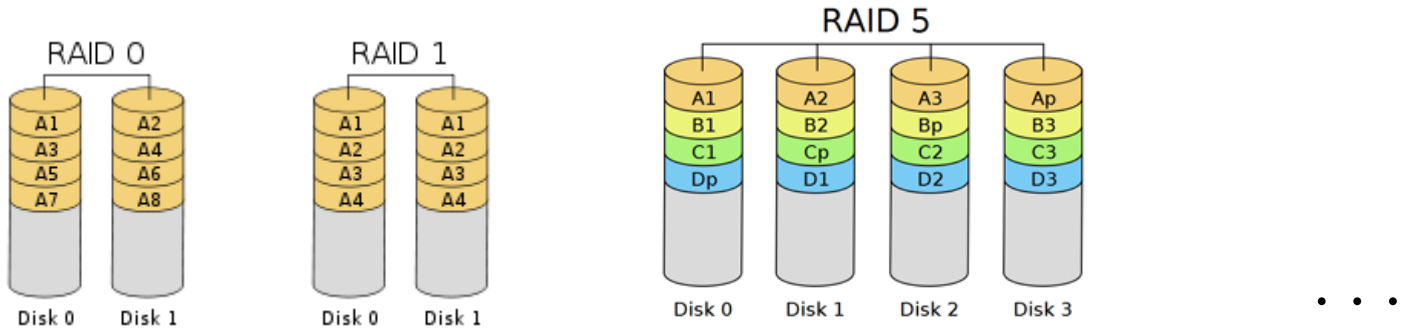
- **Move** the data
- **Store** all your original data
- Keep it **safe**
- Be able to **find** it again

Some useful things for the microscopist to know about hard-drives and servers

Drive formats:

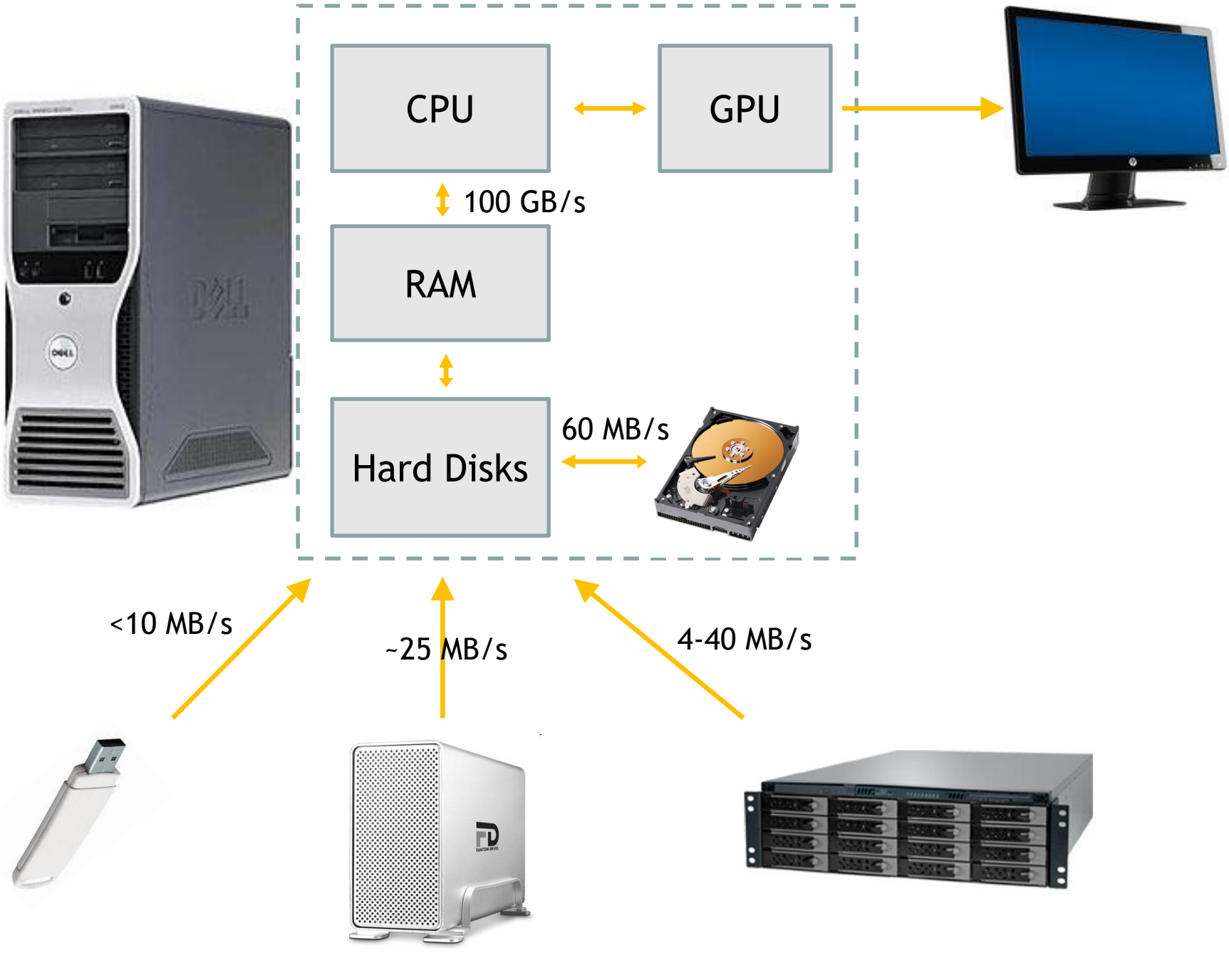
- NTFS = PC (and Linux understands)
- HFS+ = Mac OS X Extended
- Ext* = Linux
- FAT32 = All (but with some limitations – 4GB max file size)

Hard-drives fail, RAID helps (not always)



. . .

Non-redundant, faster write speeds


CPU
GPU
100 GB/s
RAM
Hard Disks
60 MB/s
<10 MB/s
~25 MB/s
4-40 MB/s

Programs for viewing your microscope images on your computer

Free basic versions of the confocal software for the particular format

(but FIJI opens them all)

Irfanview opens 16-bit TIFF (PC only)

Download links for all http://microscopy.duke.edu/analysis.html

Are we going to use Photoshop?

No

Pho·to·shop [**foh**-toh-shop]

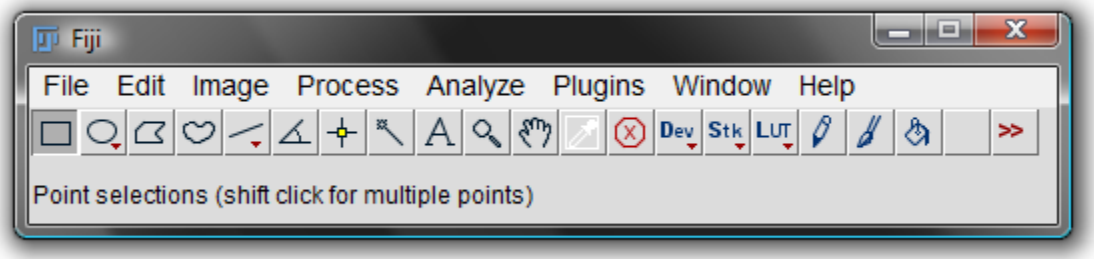
verb (used with object), **Pho·to·shopped, Pho·to·shop·ping.**
(*sometimes lowercase*) to alter (an image) using this software: *Her face is nicely Photoshopped in the ad. They've photoshopped the car onto an image of a beautiful beach.*

GNU Image Manipulation Program

http://www.gimp.org/

FIJI/ImageJ and reasons to use it

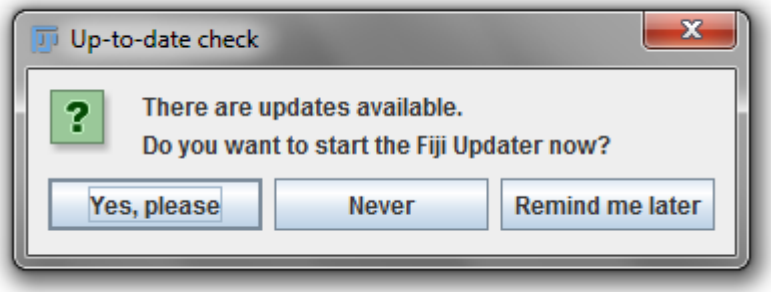
FIJI – FIJI Is Just ImageJ http://fiji.sc/wiki/index.php/Fiji

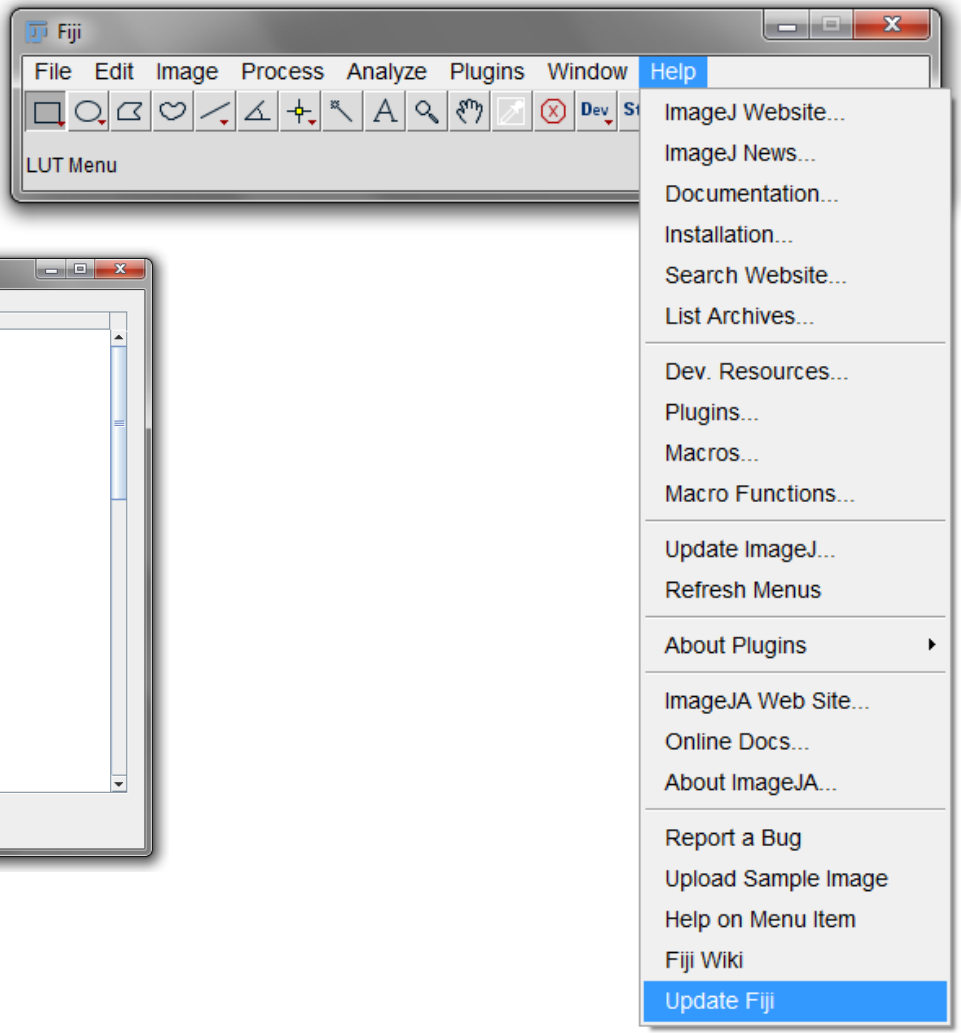


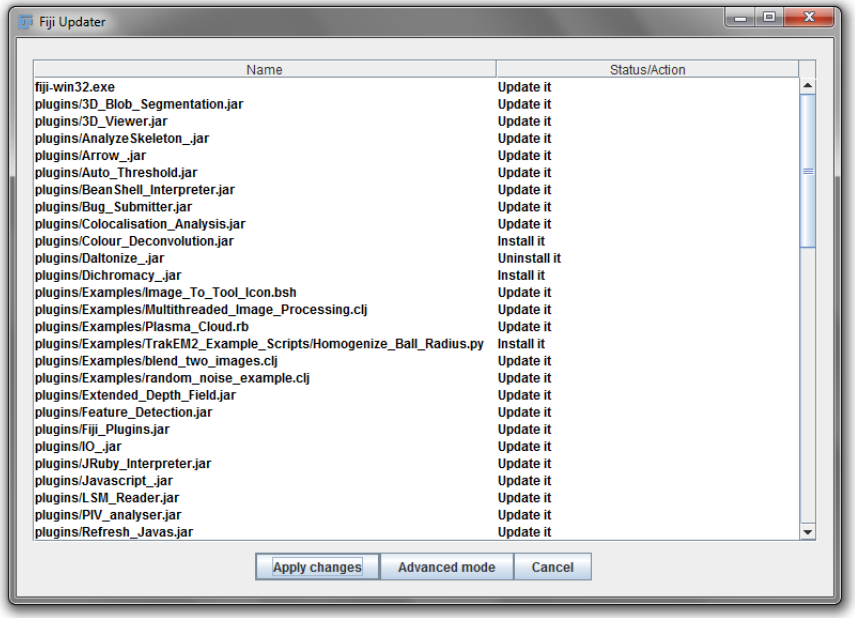
(FIJI is ImageJ + a set of useful plugins, an update tool, 3D Java, scripting and . . .)

- It's free (no cost, no licenses to deal with)
- Opens nearly anything
- It is extremely useful – many and powerful features
- It works on any OS
- Open source – (can view and modify the source code – you know what it is doing)
- Active development – lots of other people like you using it
- Has an incredible number and variety of plugins

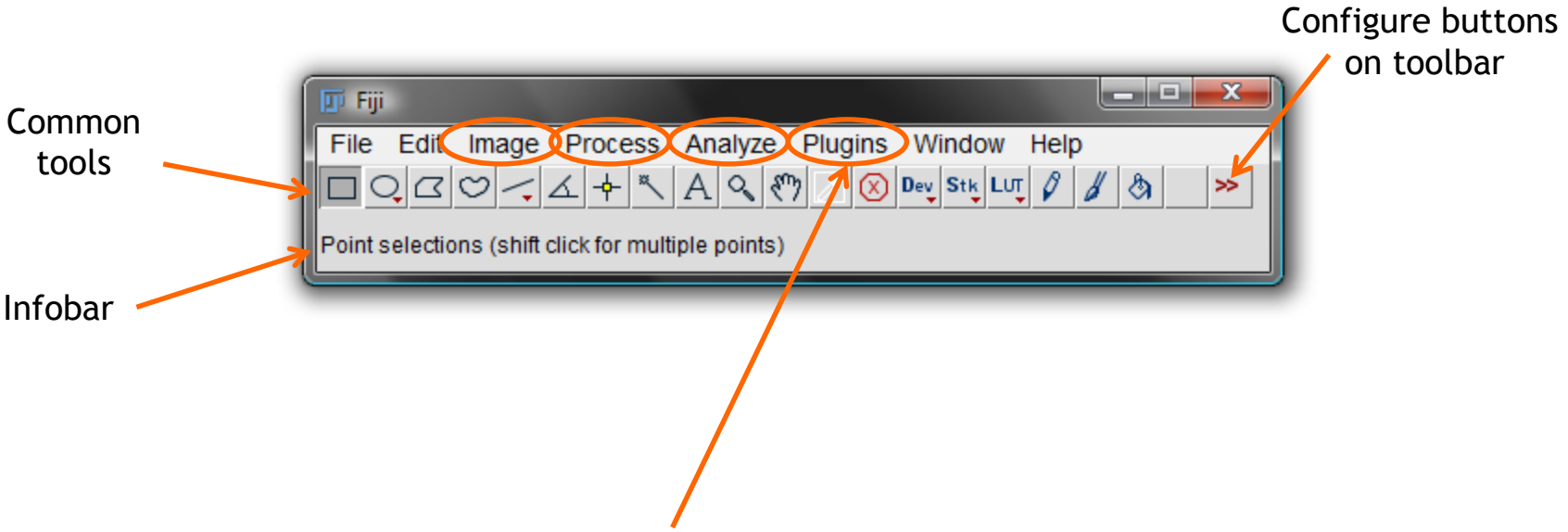
μ-Manager
The Open Source Microscopy Software


Up-to-date check
There are updates available.
Do you want to start the Fiji Updater now?
Yes, please
Never
Remind me later


Fiji
File Edit Image Process Analyze Plugins Window Help
LUT Menu
ImageJ Website...
ImageJ News...
Documentation...
Installation...
Search Website...
List Archives...
Dev. Resources...
Plugins...
Macros...
Macro Functions...
Update ImageJ...
Refresh Menus
About Plugins
ImageJA Web Site...
Online Docs...
About ImageJA...
Report a Bug
Upload Sample Image
Help on Menu Item
Fiji Wiki
Update Fiji


Fiji Updater
Name
Status/Action
fiji-win32.exe Update it
plugins/3D_Blob_Segmentation.jar Update it
plugins/3D_Viewer.jar Update it
plugins/AnalyzeSkeleton_.jar Update it
plugins/Arrow_.jar Update it
plugins/Auto_Threshold.jar Update it
plugins/BeanShell_Interpreter.jar Update it
plugins/Bug_Submitter.jar Update it
plugins/Colocalisation_Analysis.jar Update it
plugins/Colour_Deconvolution.jar Install it
plugins/Daltonize_.jar Uninstall it
plugins/Dichromacy_.jar Install it
plugins/Examples/Image_To_Tool_Icon.bsh Update it
plugins/Examples/Multithreaded_Image_Processing.clj Update it
plugins/Examples/Plasma_Cloud.rb Update it
plugins/Examples/TrakEM2_Example_Scripts/Homogenize_Ball_Radius.py Install it
plugins/Examples/blend_two_images.clj Update it
plugins/Examples/random_noise_example.clj Update it
plugins/Extended_Depth_Field.jar Update it
plugins/Feature_Detection.jar Update it
plugins/Fiji_Plugins.jar Update it
plugins/IO_.jar Update it
plugins/JRuby_Interpreter.jar Update it
plugins/Javascript_.jar Update it
plugins/LSM_Reader.jar Update it
plugins/PIV_analyser.jar Update it
plugins/Refresh_Javas.jar Update it
Apply changes
Advanced mode
Cancel

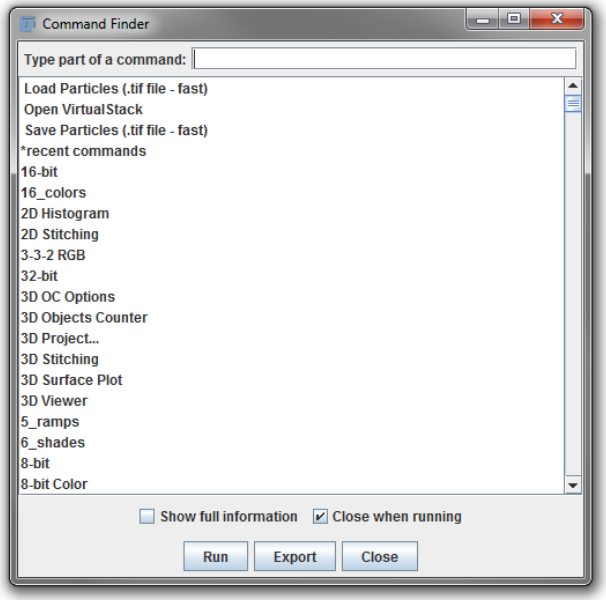
Overview of the FIJI software



Plugins and Macros are extensions of core functions – write them yourself or use ones written by other people

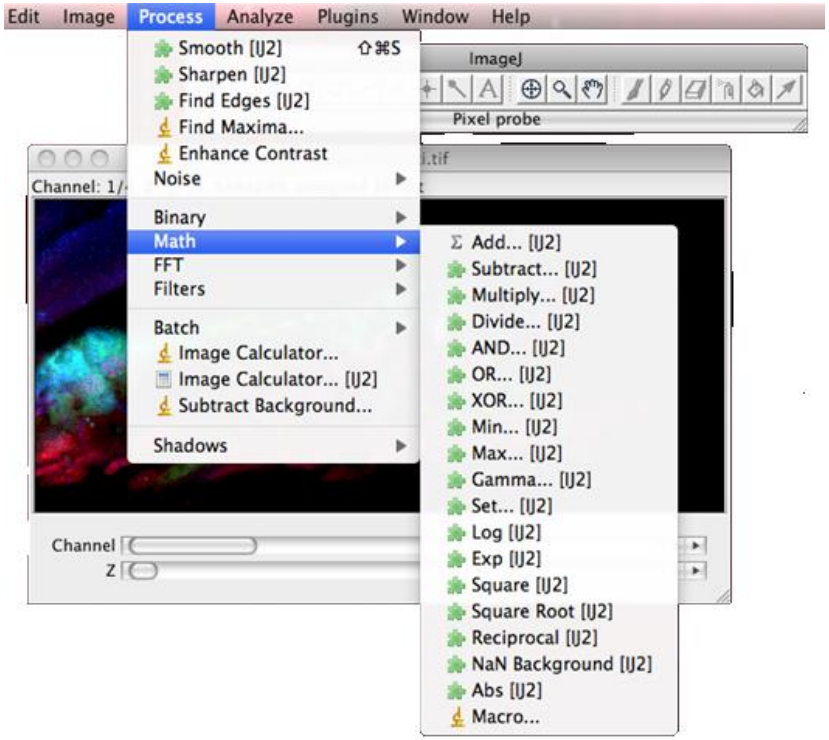
Oddities of the FIJI software

- Multiple Document Interface – lots of little windows
  File/Close All is quite useful
- Java based – System and Java clipboards
- Undo – doesn't always work – *File/Revert (ctrl-R)*
- *Ctrl-L* gives command Finder

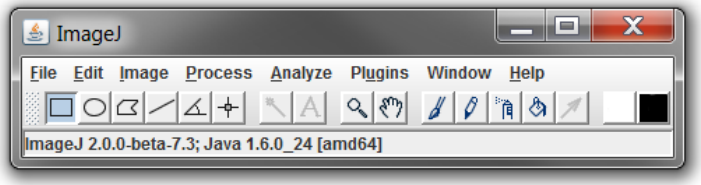


ImageJ2

http://developer.imagej.net/downloads



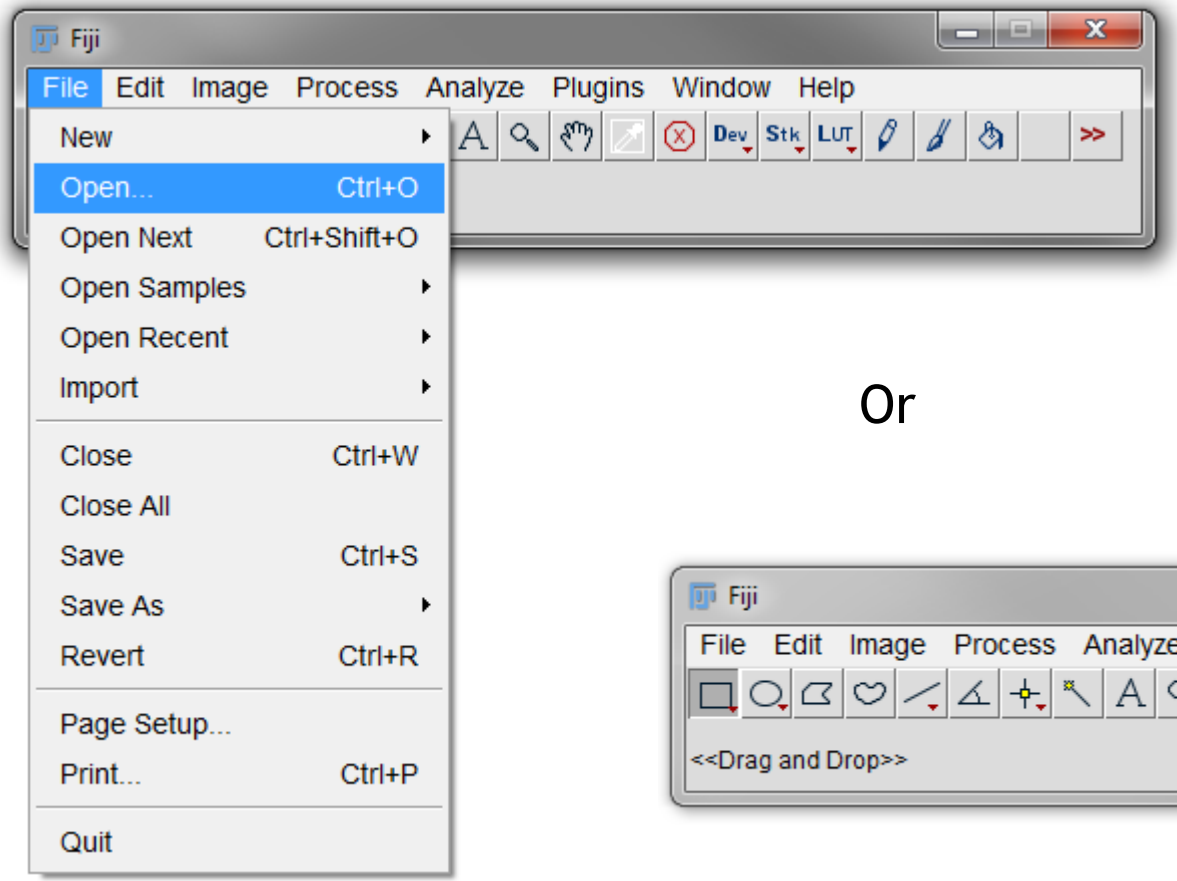
Help/Switch To Modern Mode



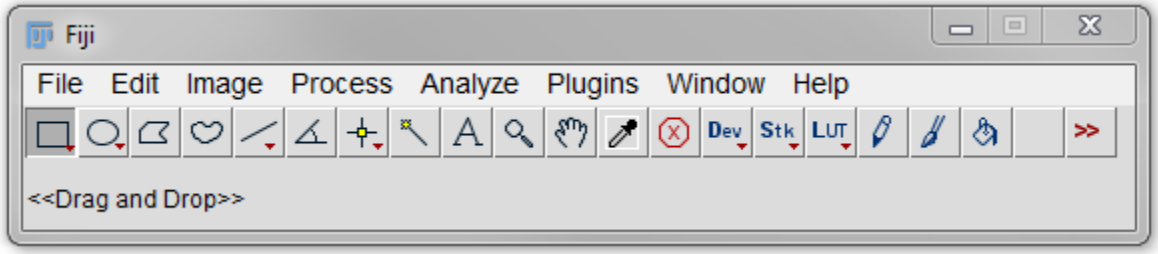
Headless | N-dimensional formats | Complete re-write

Opening a file

File:
FirstImage.tif



Or



Really basic functions

+ = Zoom in

- = Zoom out

5 = 1:1 zoom

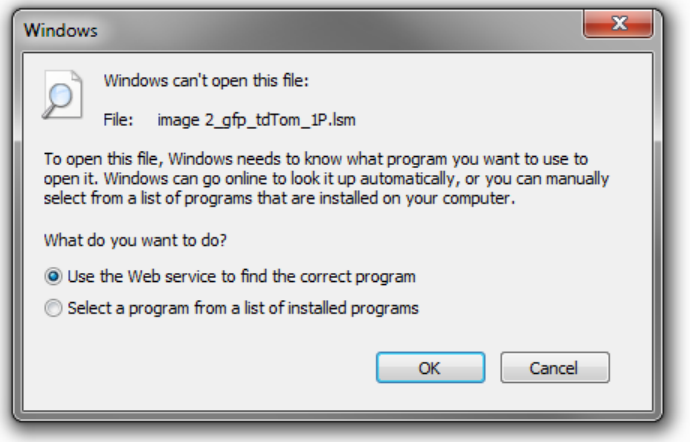
Hold space bar and left click drag = Pan

Duplicate the image:
/Image/Duplicate (or ctrl+shift+D on a PC)

Images are in many formats

Microscope specific formats

LIF, LEI, LSM, ZVI, STK, OIB, OIF, dv, ICS/IDS, ICS2, r3d, PIC, IPL, CXD, SDT, IPW, ND2, RAW, IMS. . .



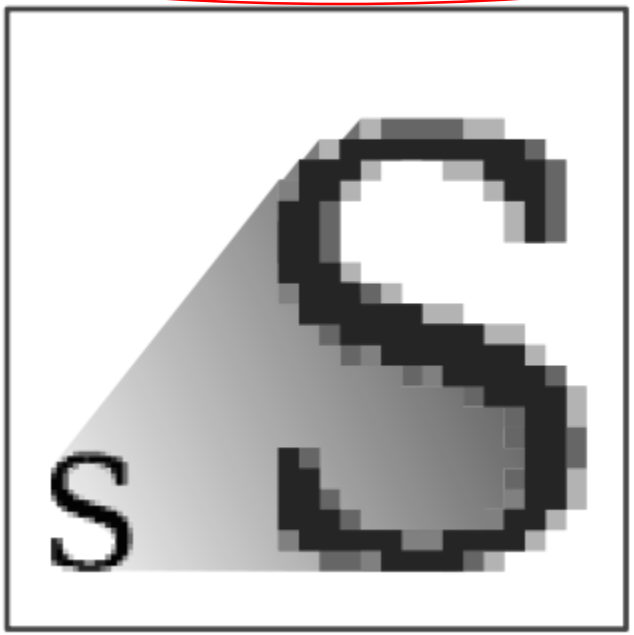
General formats

TIF, GIF, BMP, PNG, SVG, JPEG, MPEG, QuickTime, AVI, SWF, PICT, EPS, PDF, PSD, XCF . . .

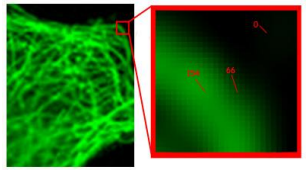
Which of these are good and what characteristics do they have?

Two types of images

Bitmap/raster



Array of pixels



Photoshop/GIMP

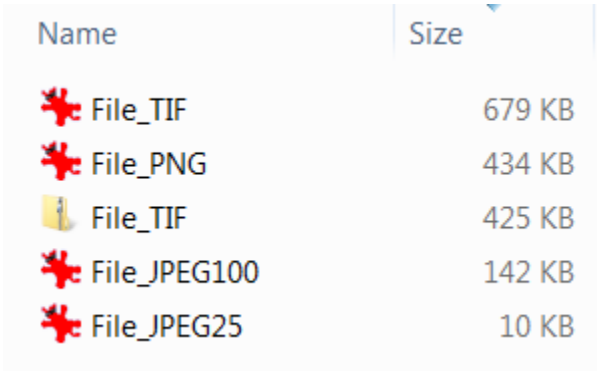
Vector

Mathematical/geometric equations
Scalable
Efficient for some things

Illustrator/Inkscape

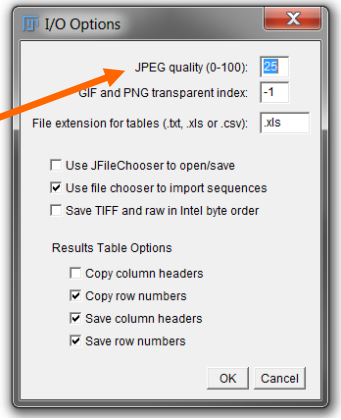
Compression

Files:
These ones



- How different do the images look?
- Which ones are wise to use?

(I made the JPEGs different using Edit/Options/InputOutput

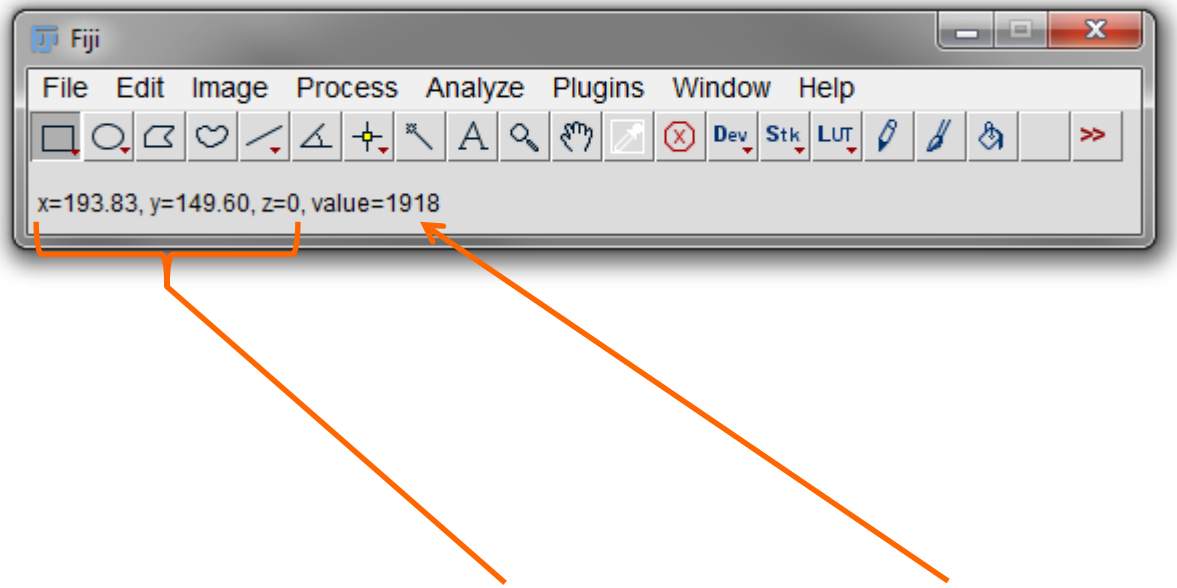


Compression types

Lossless	Lossy
Encodes exactly the original data (encodes redundant information and some other clever tricks)	Throws some information away to reduce the file size
You can get back the original	You can not get the original back
Sometimes not that much smaller	Allows a much smaller file, Range of quality/size
Zip etc (takes time), LZW	
PNG	JPEG

Intensity range and displaying them optimally and fairly

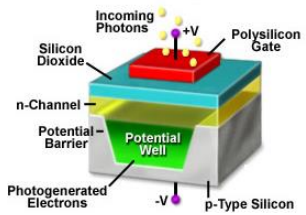
What intensities are in an image?



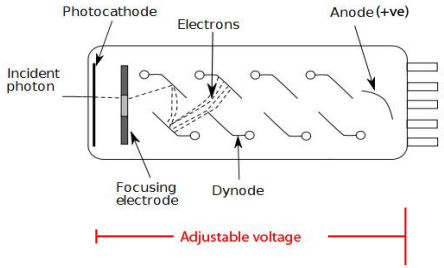
Mouse over for: Coordinates | Intensity

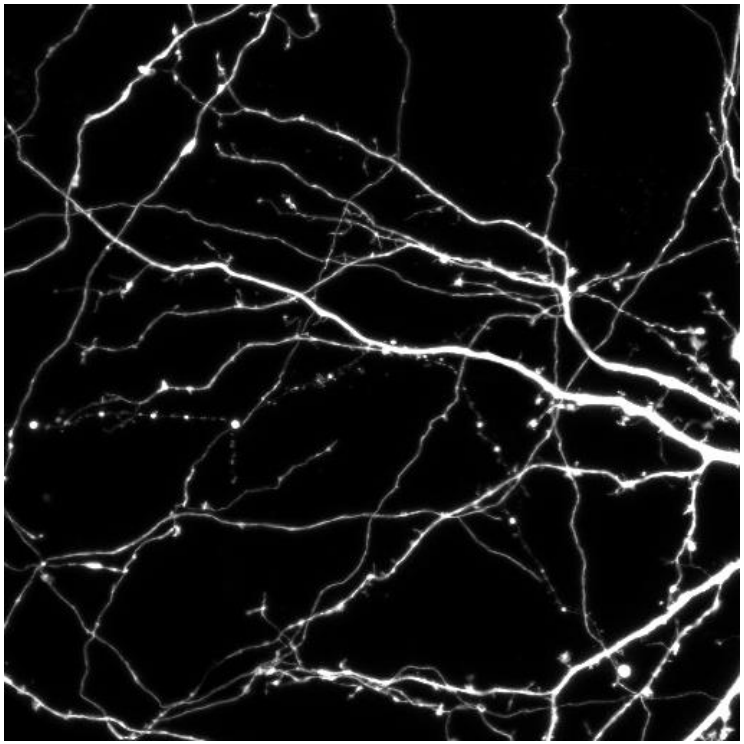
Analog to digital conversion

CCD

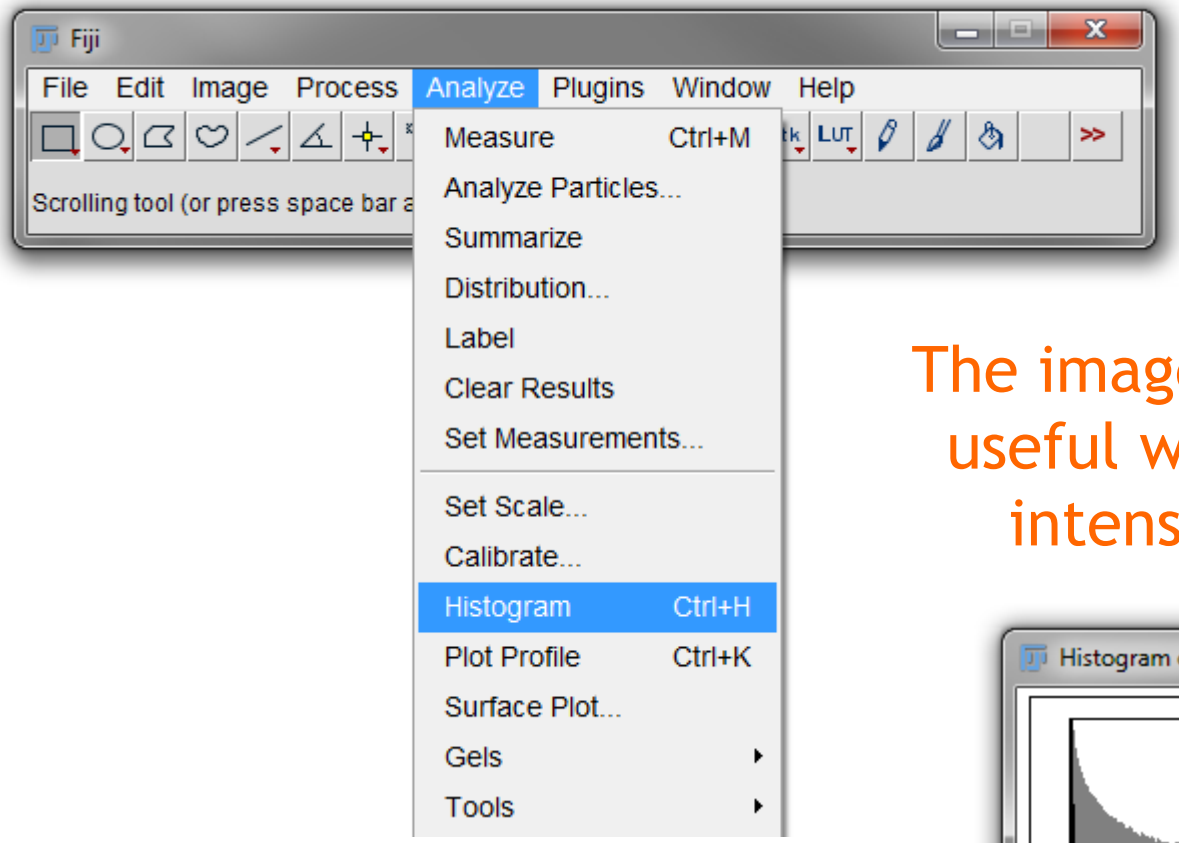


PMT

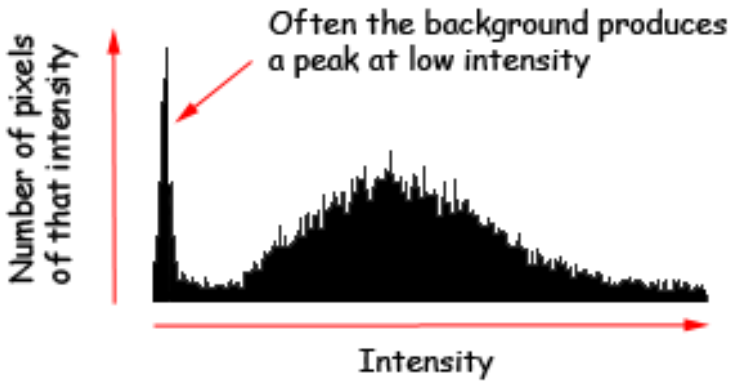


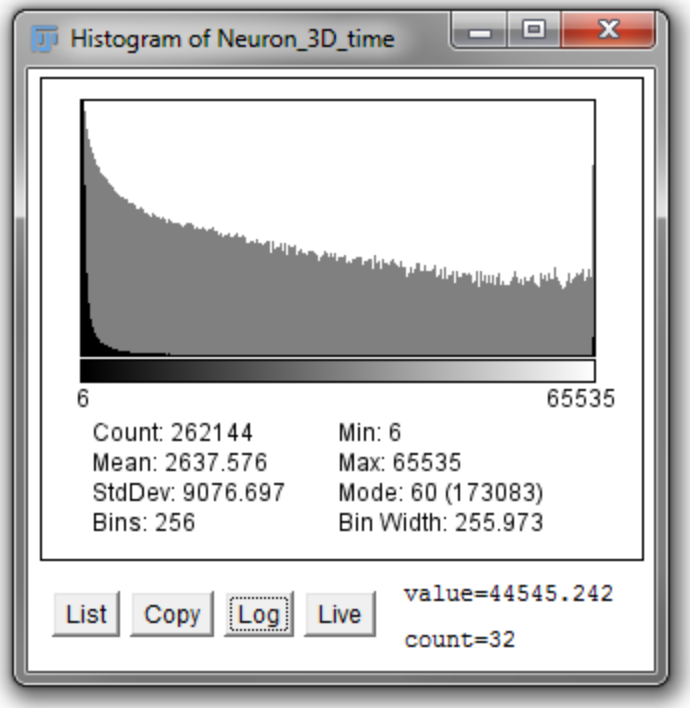


Histogram



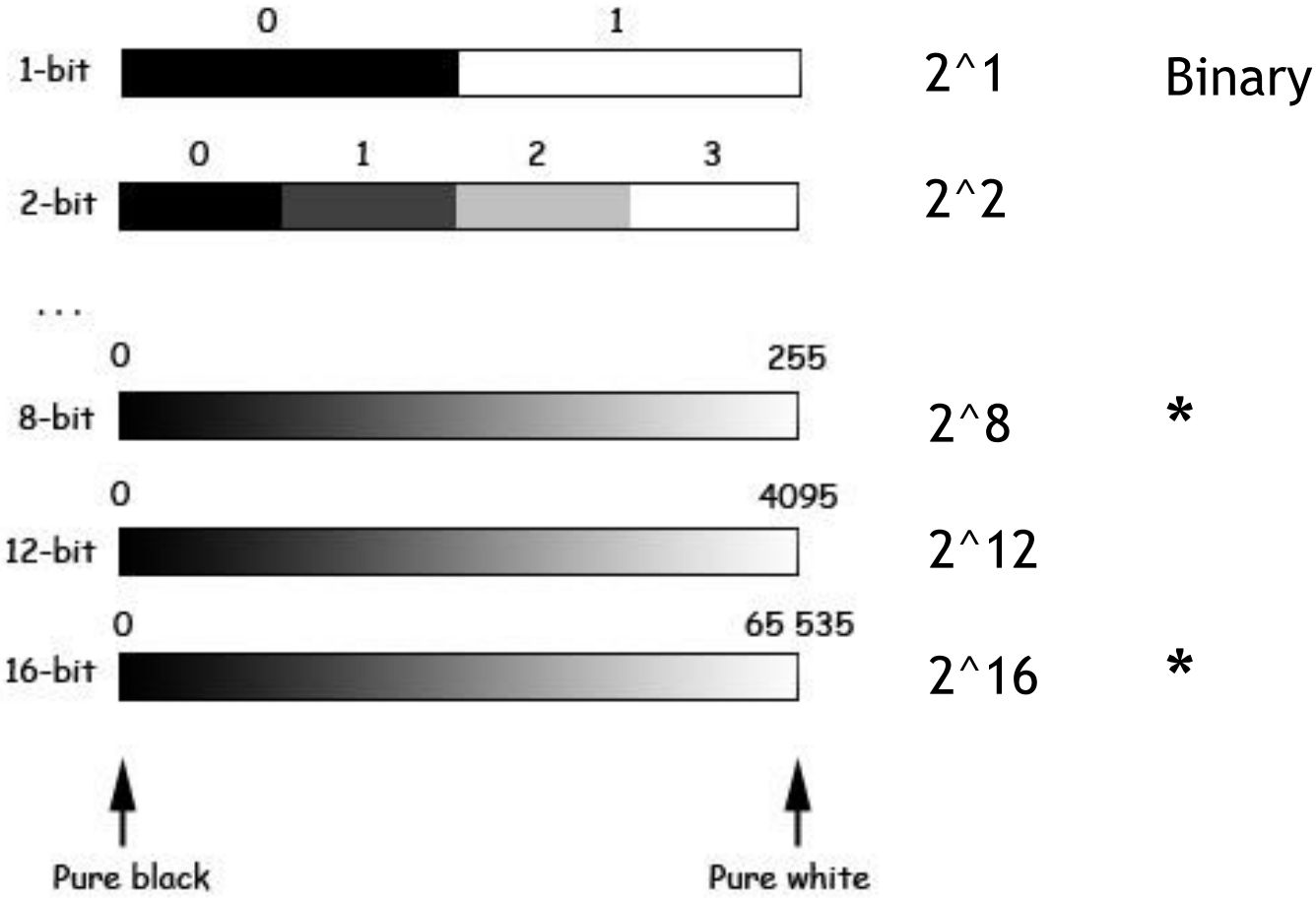
The image histogram is a very useful way of looking at the intensities in the image





Bit-depth

The range of intensities can be represented with different numbers of gray-levels



Even more Bit-depths

- Signed and unsigned 16 bit
  - 0-65,535
  - -32,768 to + 32,767

(What is a negative intensity?)

- 32-bit float – 4.3 billion + Decimals + NotaNumber

How many grey levels do we need?

8-bit enough for most things:

- Human eye (without adaptation)
- Monitors – contrast factor
- Printers

But it is useful to use more in microscopy?

Dynamic range

What span of a measurement can be measured?

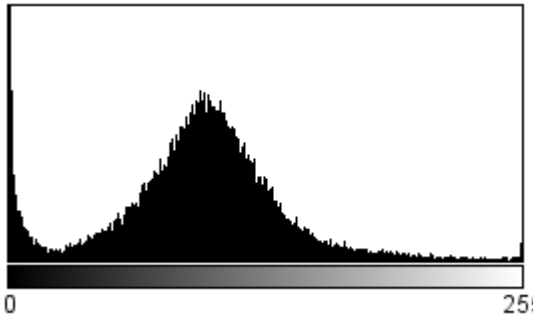
$$\frac{\text{Signal at the greatest intensity of light measurable}}{\text{Lowest intensity (Noise limited)}}$$

How many grey levels do we need?

How many photons/px are involved?

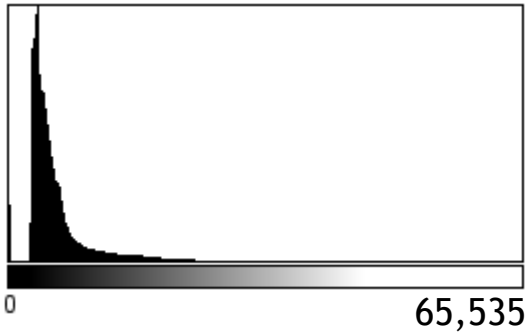
Say, <100 ~1000 10-100 before EM

Confocal

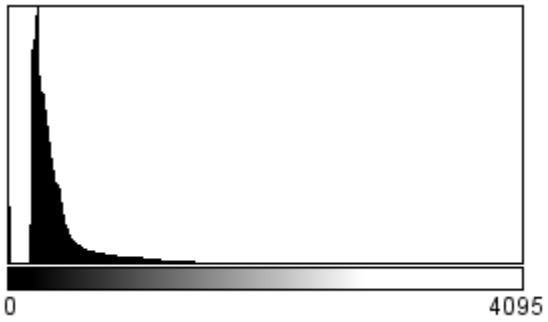


CCD

EM-CCD

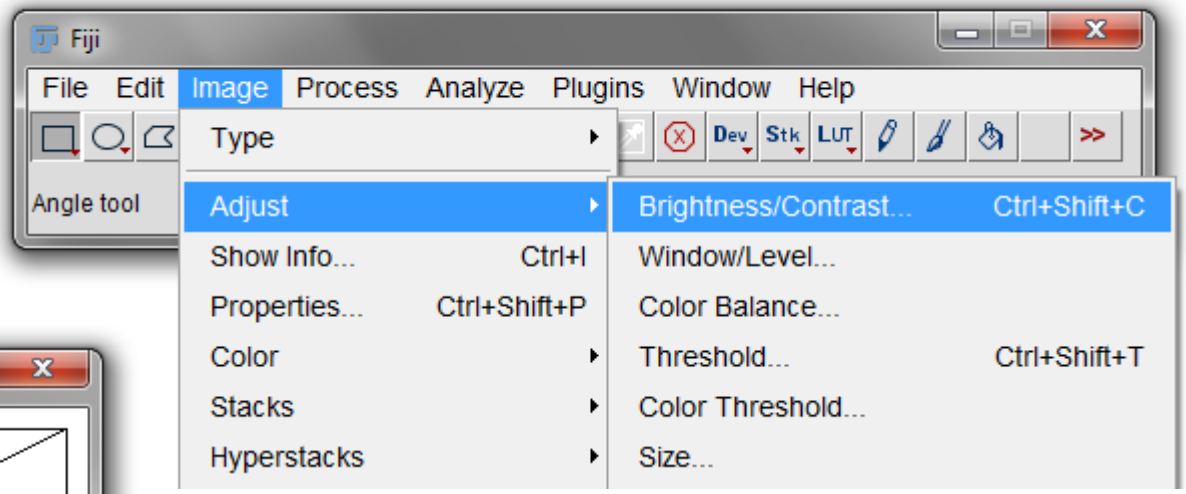


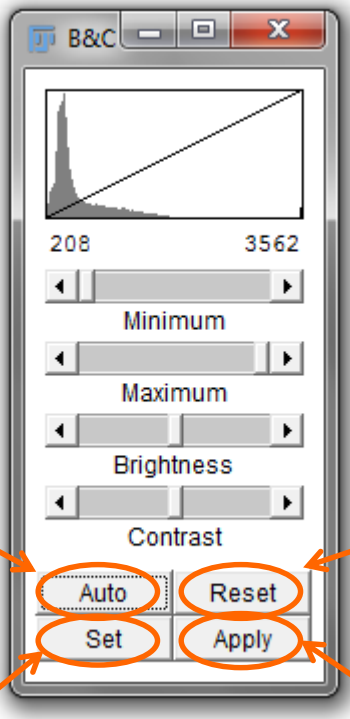
What if only some of the range is filled?



Scaling

File:
B_Imageforscaling





Enhance contrast

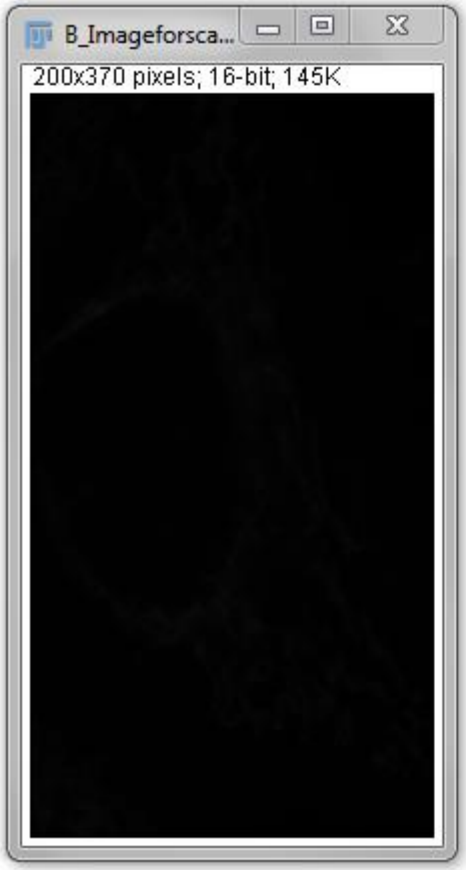
Min to Max

Type in numbers, can set to all open images

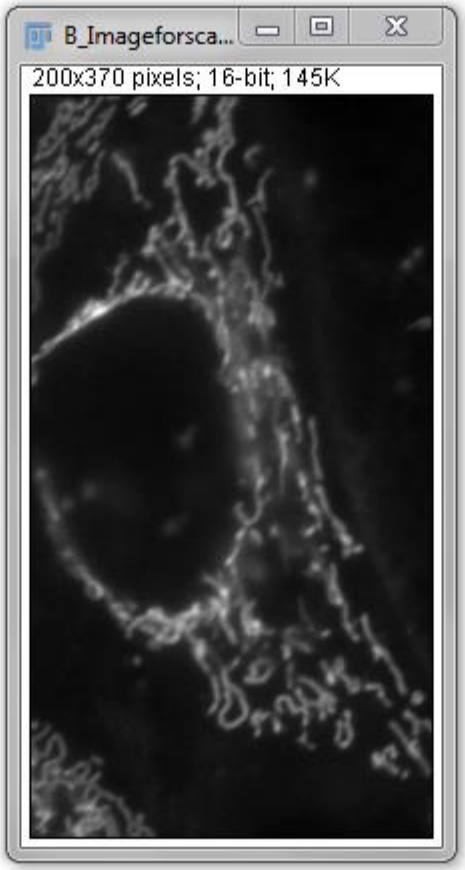
<u>Changes the pixels!</u>

Contrast in image scaling

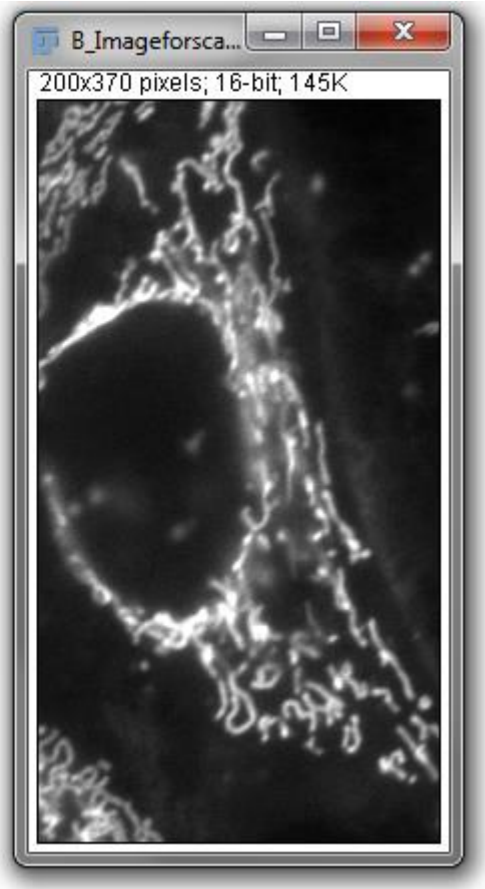
Full range of CCD



Full range of image



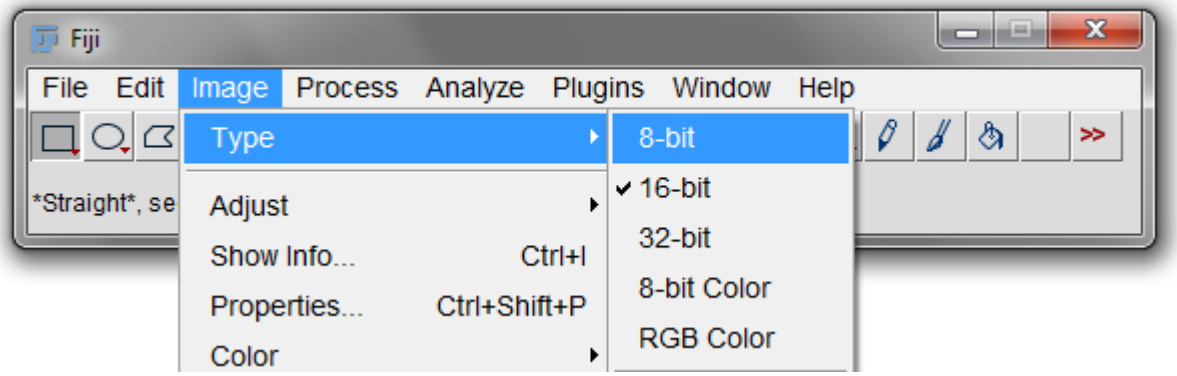
Slight clipping



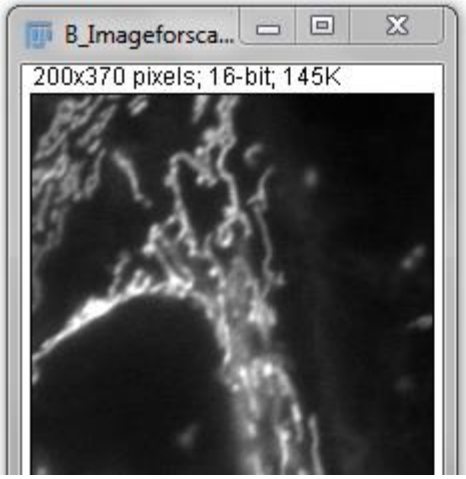
Why is this useful,
when is this ok,
how much is ok?

Exporting an 8-bit copy

Ctrl+shift+D
=Duplicate

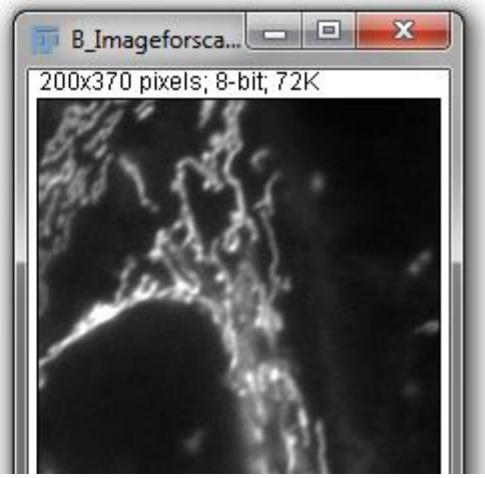


Save your original images – use for quantification



Original pixel intensity range stretched or compressed to 8-bit

Easy display/publication



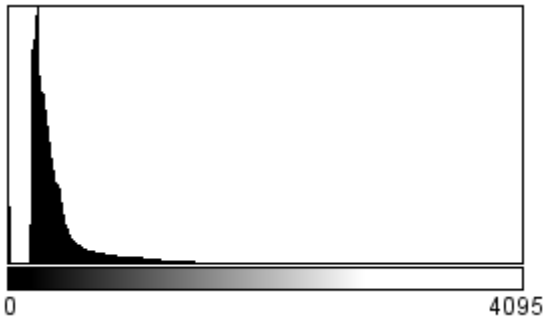
Which format should I use for export?

Probably makes sense to use the format of the microscope as much as possible, if you do need to export to a standard format . . .

TIFF (aka TIF), and variants

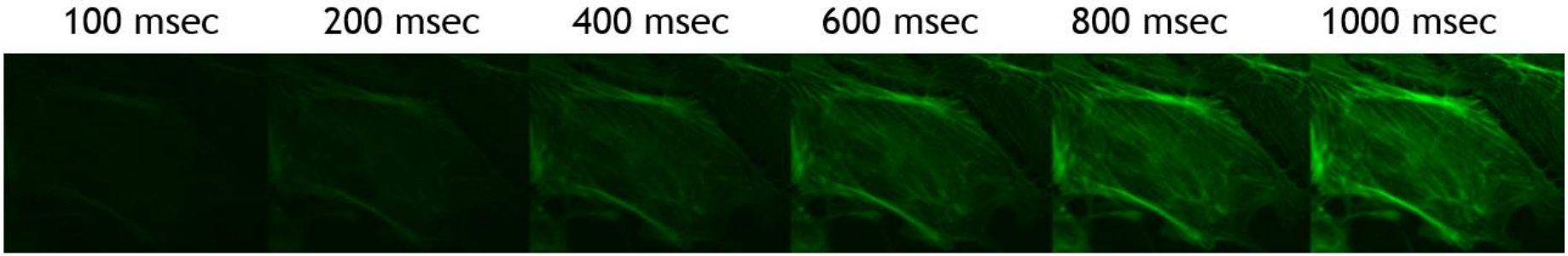
- Lossless
- Suitable bit-depths (8-bit or 24-bit RGB for export)
- Space for metadata (not always populated properly)
- Open
- Pretty much universal

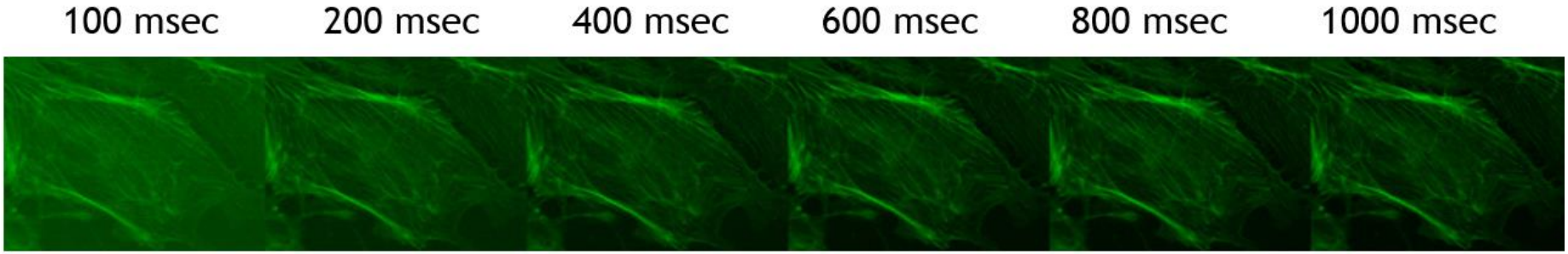
What happens to the numbers?



Scaling has limitations

Scaled to 0 - 4095

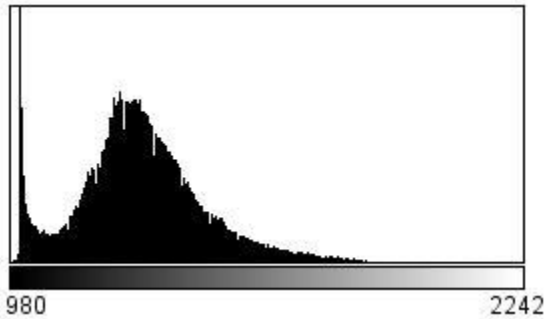


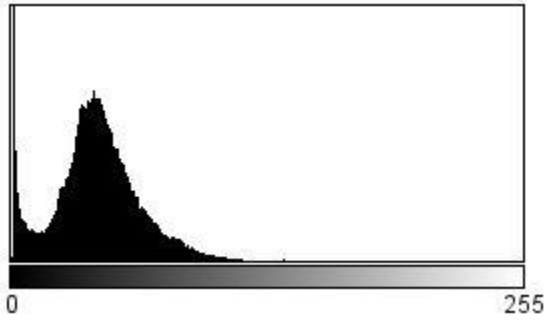


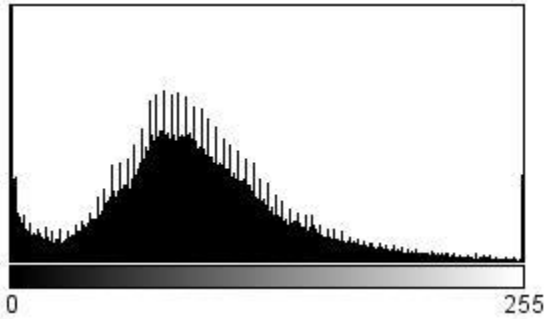
Auto-scaled to min-max of each image

How many grey-values should you have in your raw images?

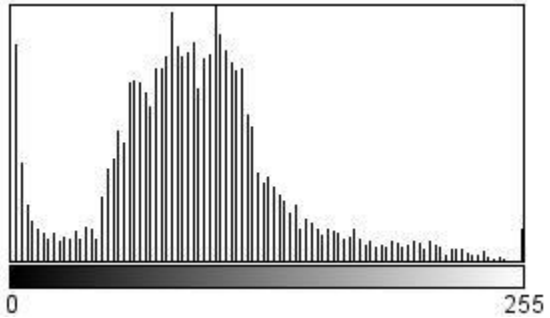
Imperfections in image scaling







12-bit to 8-bit compression

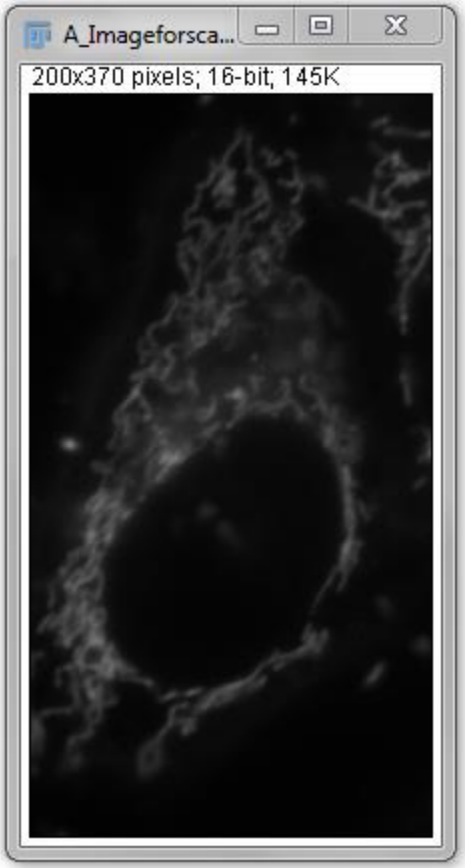


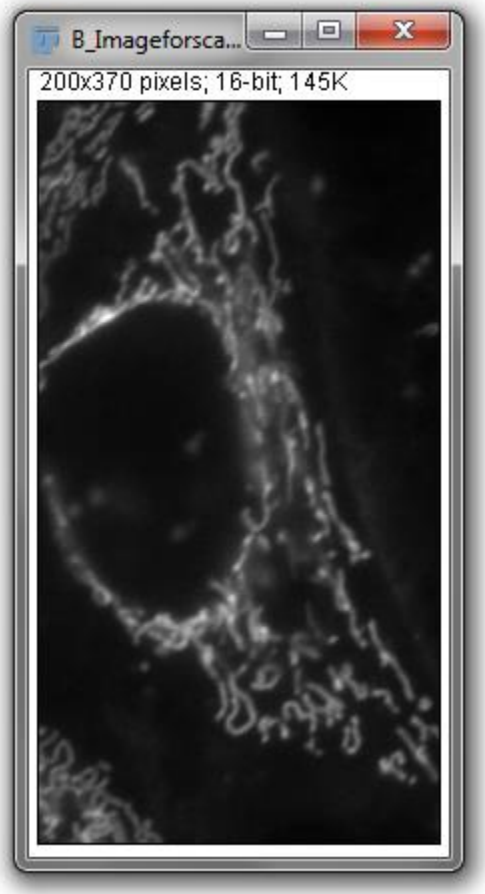
8-bit to 8-bit Stretch

Avoid scaling any images you want to quantify

Exercise: Fair display

Files:
A_Imageforscaling
B_Imageforscaling





Display these to:

1. Reflect the intensities
2. Show best structure

Make 8-bit copies of each pair

Inverting an image: making a negative

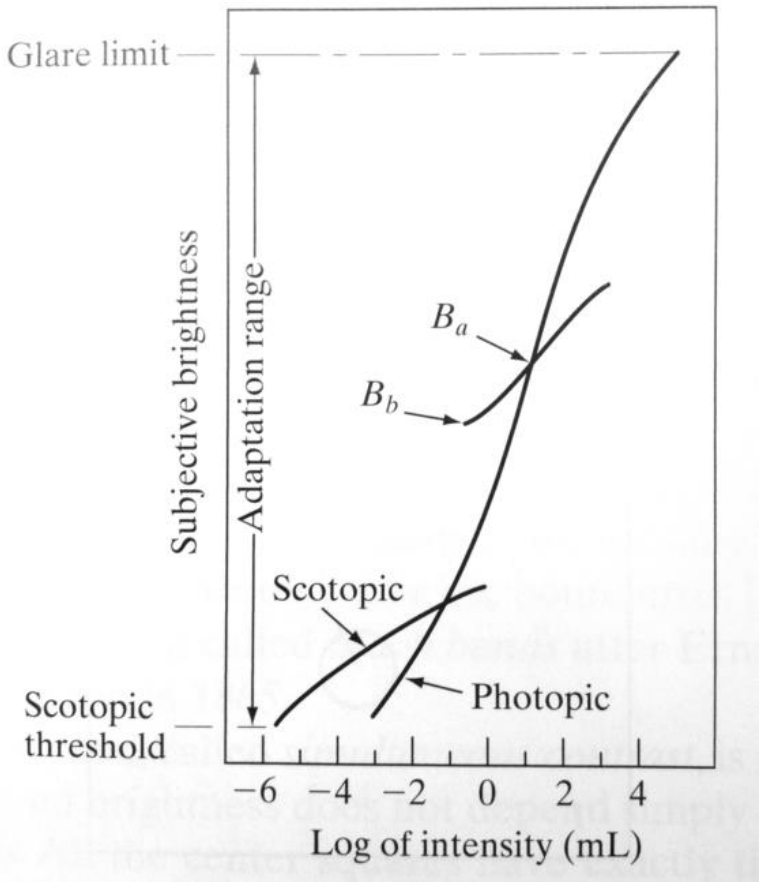
Files:
Invert_this.tif
A_Imageforscaling

- Try inverting an image: Edit/Invert
- Look at the histogram
- Mouse-over the pixels
- Scale it
- Compare to original

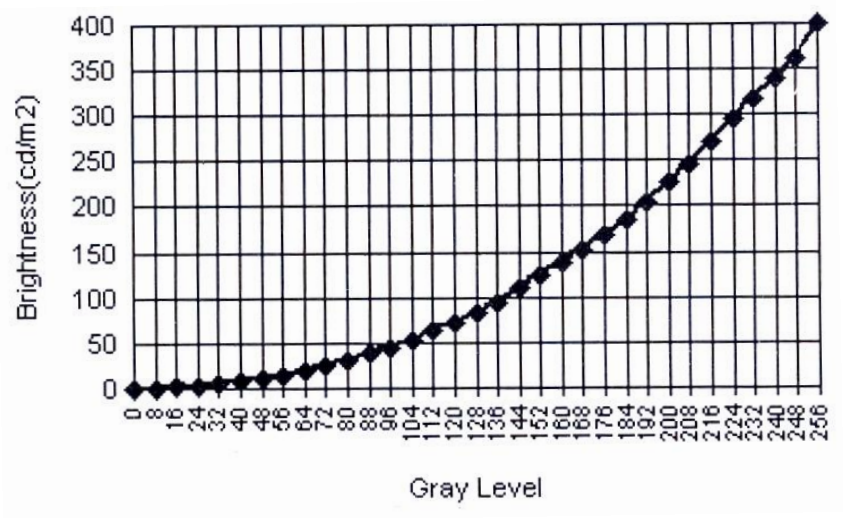
Linearity in intensity display

That sounds a good idea, let's do that

Problem



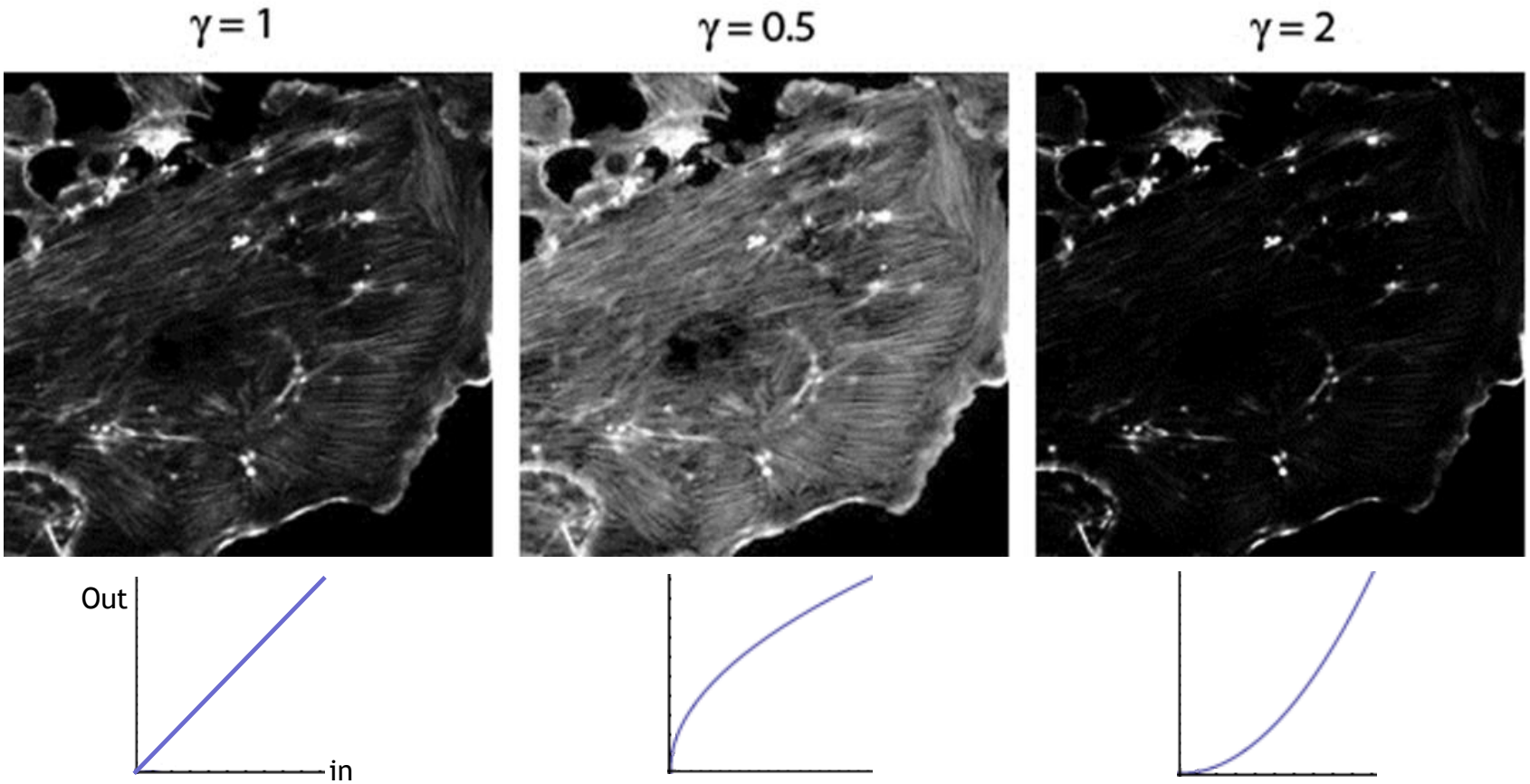
Solution



Gamma

File: gamma

$$\text{output} = \text{input}^{\gamma}$$



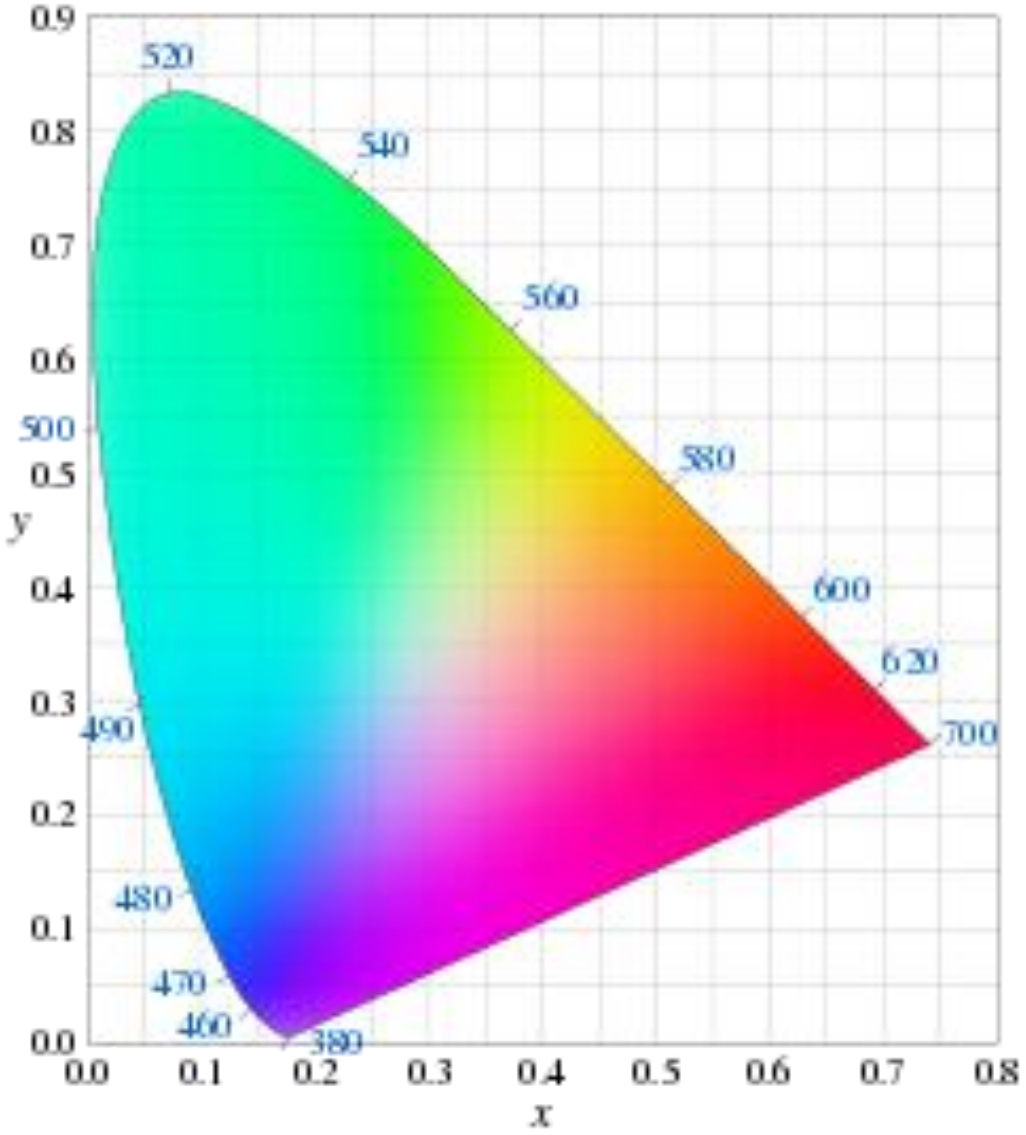
"Linear" display of brightness: $\gamma = 1$

Stick with 1 if you can, certainly state if any images are not linear

You can **change** your image: */process/math/gamma*

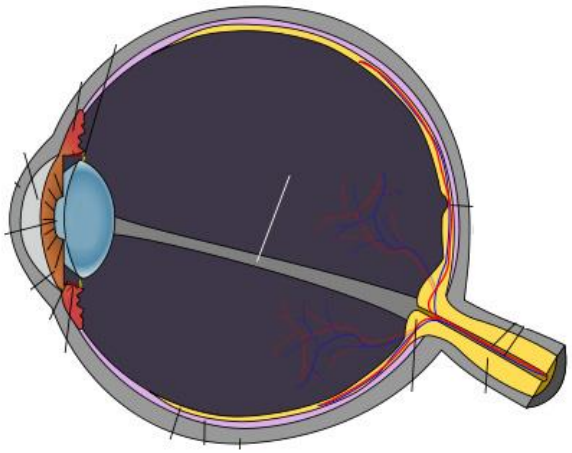
Color

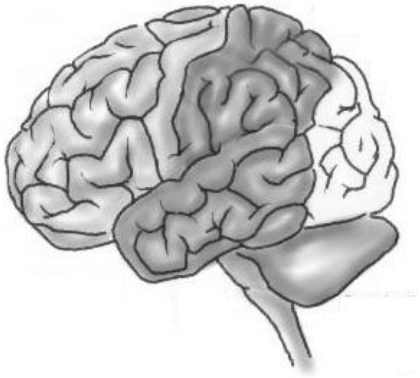
The 10 million colors we can see

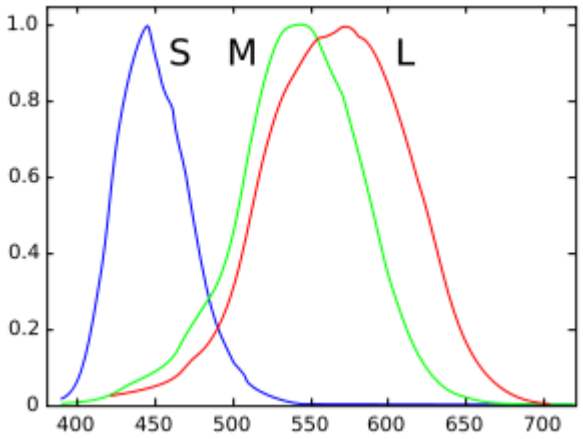


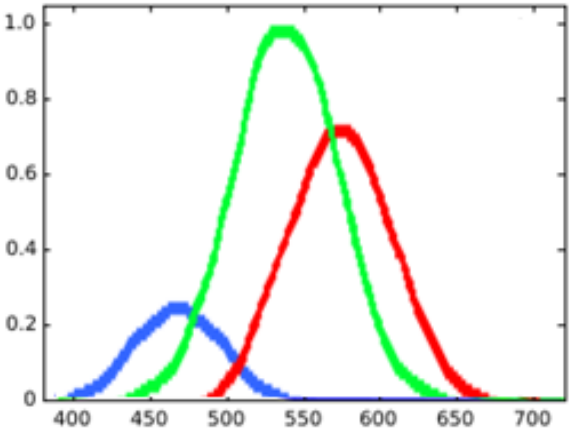
Our eyes

How many Mpx are these?



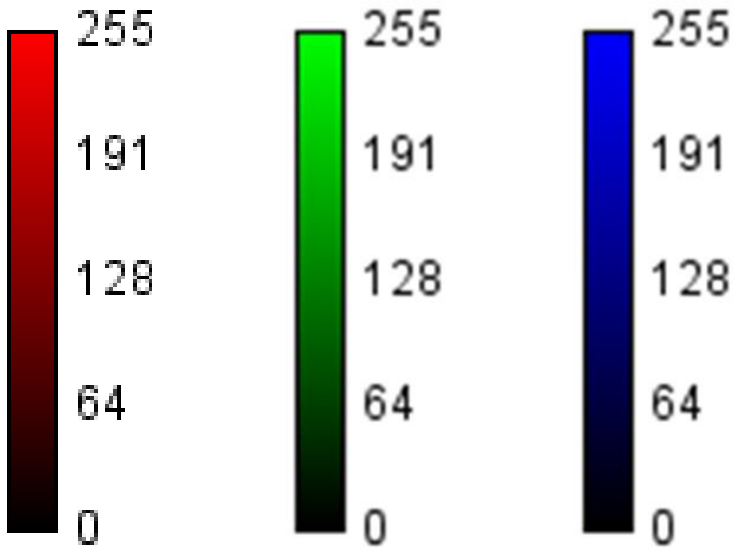






Monitors etc fit in with our eyes – tiny little R, G or B pixels

24-bit RGB

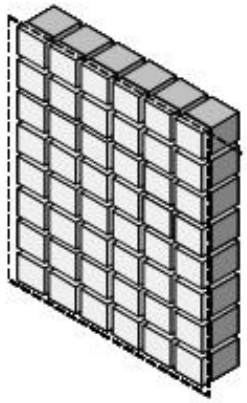


256^3 = 16.77 million

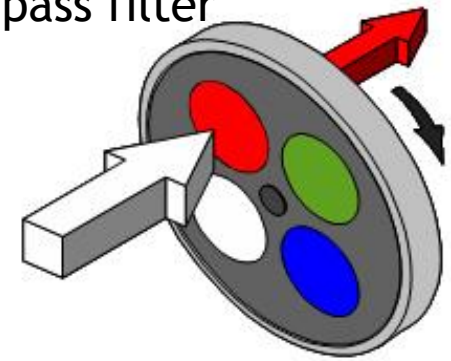
Color in imaging

Most of it is actually pseudo-color in fluorescence:

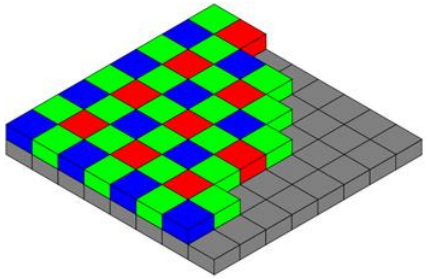
Sensitive monochrome detector



Bandpass filter

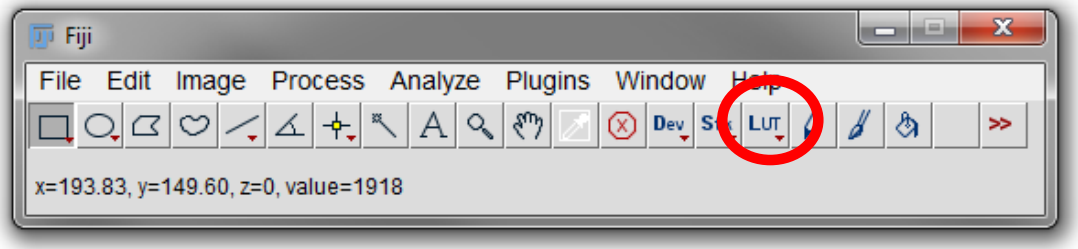


Bayer mask in color camera

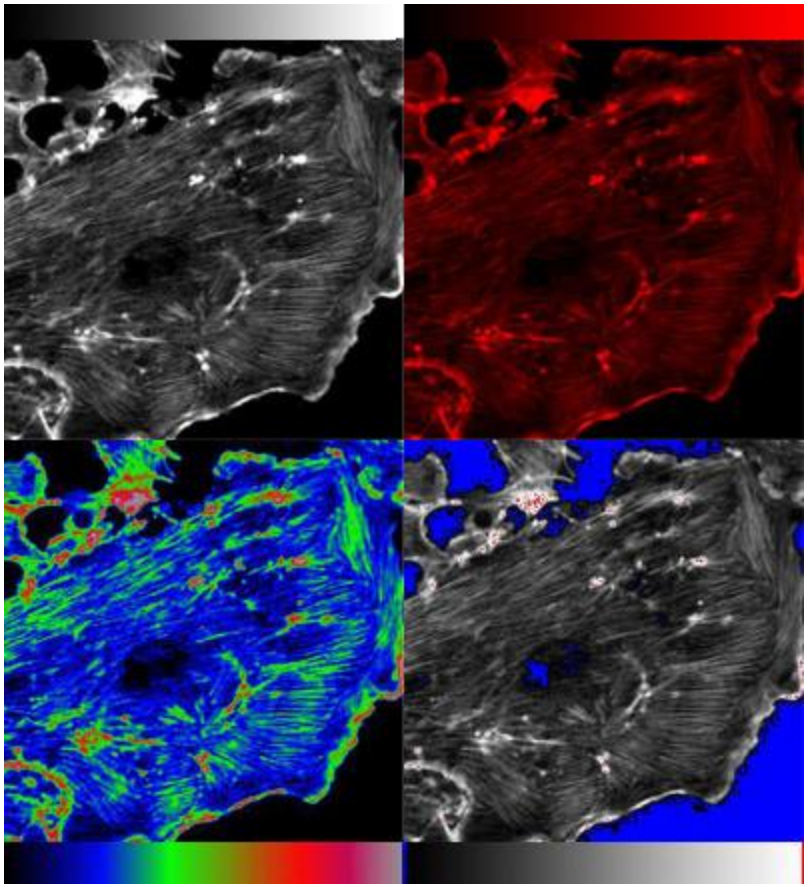


We use color to add context and aid interpretation

Lookup tables



Different ways of displaying the same image . . .



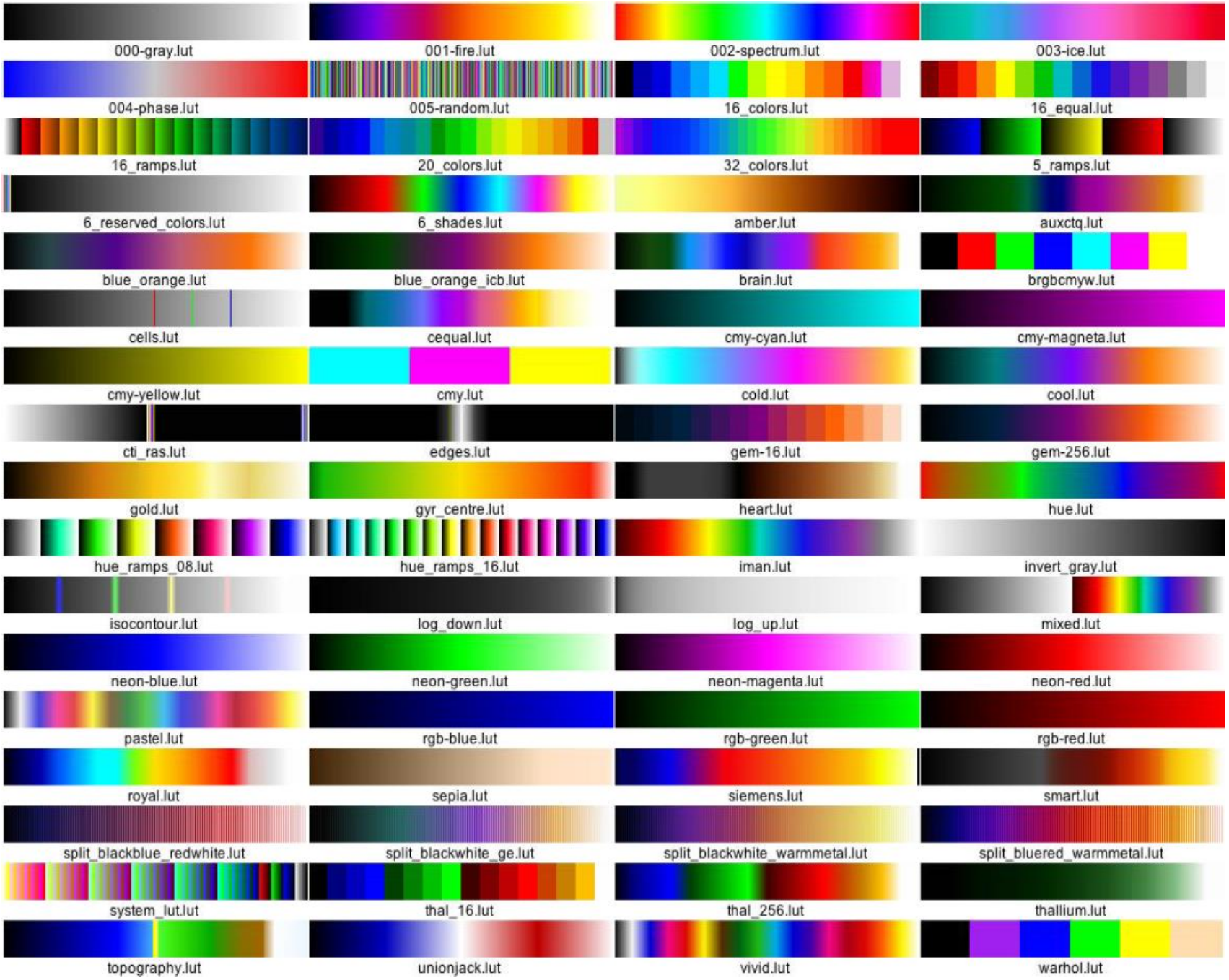
Helpful for distinguishing similar intensities

Good for adjusting the gain and offset on the confocal

Try some LUTs

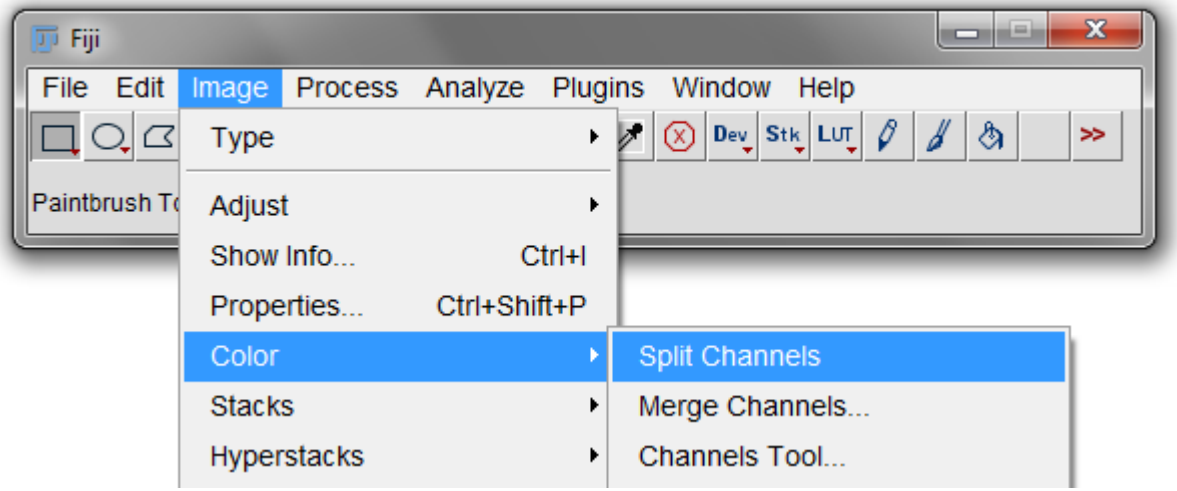
File: gamma

- Try lots of different ones
- Analyze/Tools/Calibration bar
- Inverted LUT – same as image inversion? Mouse over.
- Do you think inverted is a good way of displaying?


000-gray.lut
001-fire.lut
002-spectrum.lut
003-ice.lut
004-phase.lut
005-random.lut
16_colors.lut
16_equal.lut
16_ramps.lut
20_colors.lut
32_colors.lut
5_ramps.lut
6_reserved_colors.lut
6_shades.lut
amber.lut
auxctq.lut
blue_orange.lut
blue_orange_icb.lut
brain.lut
brgbcmyw.lut
cells.lut
cequal.lut
cmy-cyan.lut
cmy-magneta.lut
cmy-yellow.lut
cmy.lut
cold.lut
cool.lut
cti_ras.lut
edges.lut
gem-16.lut
gem-256.lut
gold.lut
gyr_centre.lut
heart.lut
hue.lut
hue_ramps_08.lut
hue_ramps_16.lut
iman.lut
invert_gray.lut
isocontour.lut
log_down.lut
log_up.lut
mixed.lut
neon-blue.lut
neon-green.lut
neon-magenta.lut
neon-red.lut
pastel.lut
rgb-blue.lut
rgb-green.lut
rgb-red.lut
royal.lut
sepia.lut
siemens.lut
smart.lut
split_blackblue_redwhite.lut
split_blackwhite_ge.lut
split_blackwhite_warmmetal.lut
split_bluered_warmmetal.lut
system_lut.lut
thal_16.lut
thal_256.lut
thallium.lut
topography.lut
unionjack.lut
vivid.lut
warhol.lut

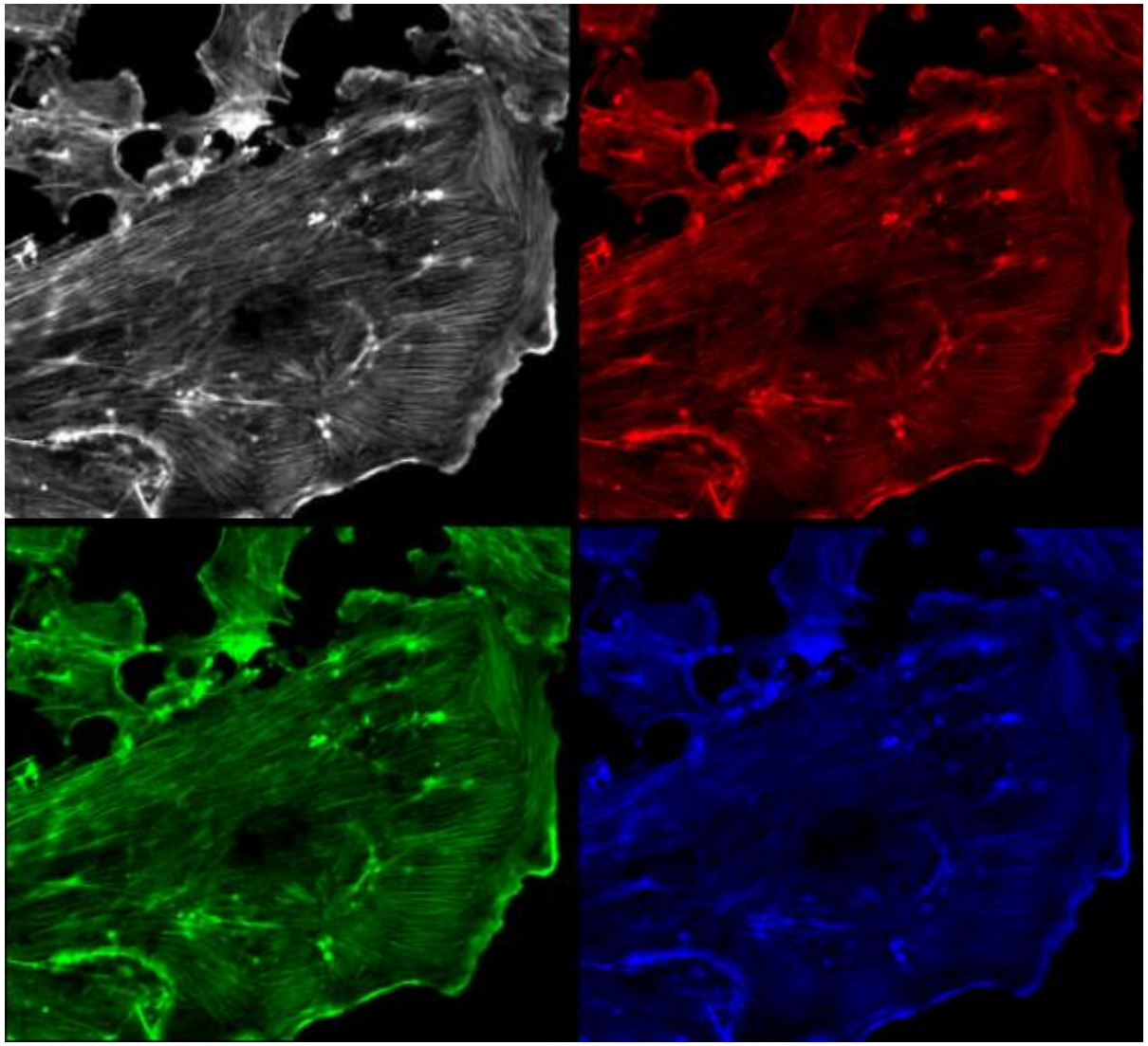
Color display

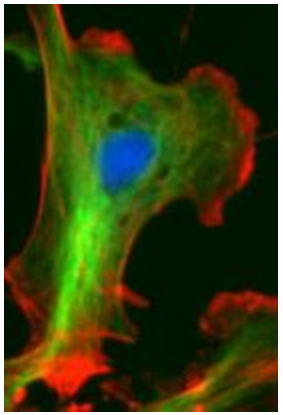
File:
FluorescentCells



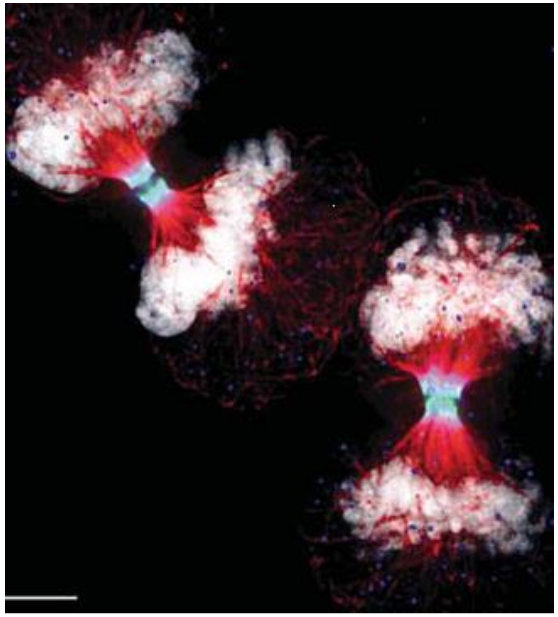
- ✓ RGB and Composite (more channels, more flexible)
- ✓ Channels Tool
- ✓ Split and Merge

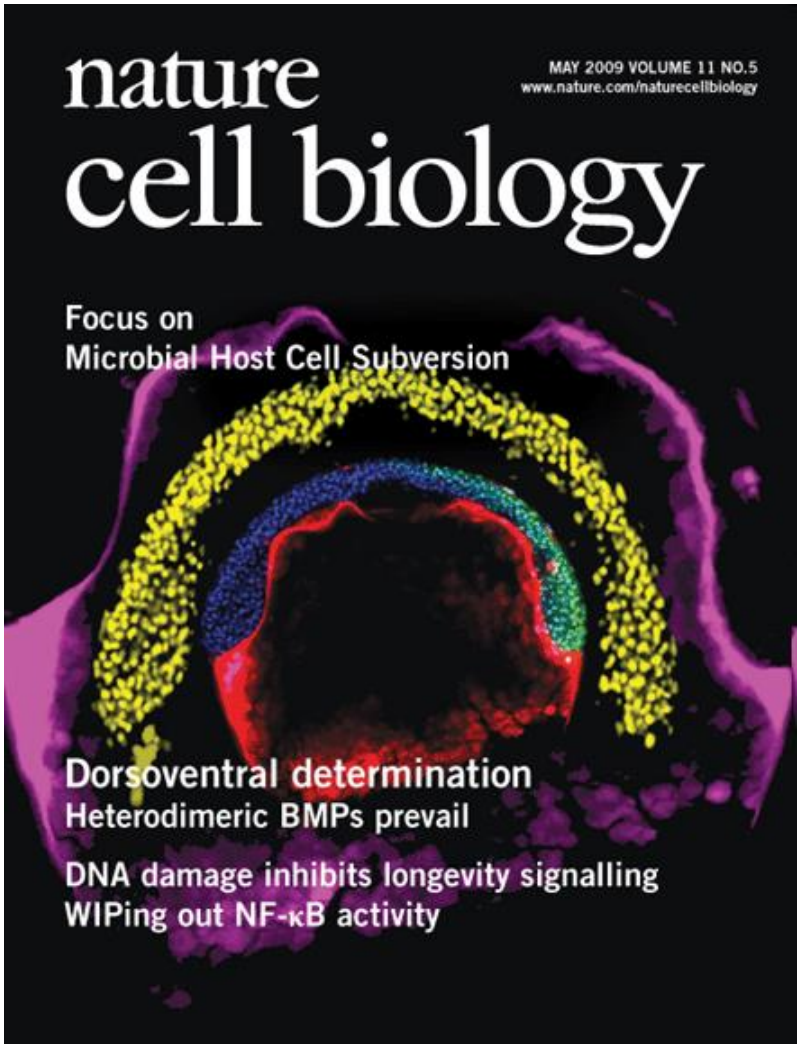
Color display isn’t always useful





4 or more color images





Exercise: Color display

- ✓ RGB and Composite (more channels, more flexible)
- ✓ Channels Tool
- ✓ Split and Merge

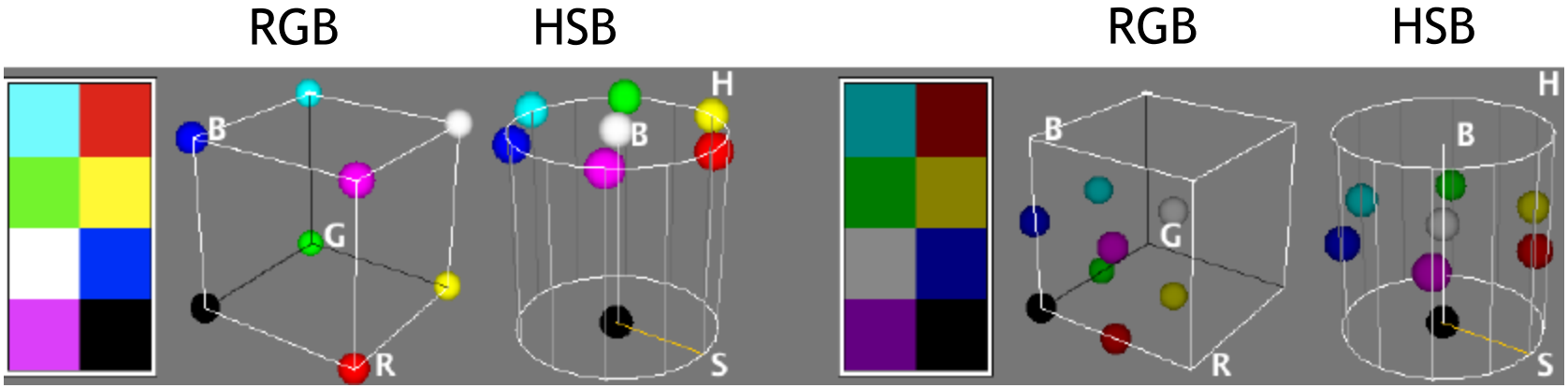
Files:
ImageQuad_A
ImageQuad_B
ImageQuad_C
ImageQuad_DAPI

Take these 4 channels of the same cell, make a composite with the channels pseudo-colored and scaled as you think best displays them and export as 24-bit RGB tiff

Should you scale the 4-channels in the same way?
Can you split the final RGB back to your 4 images?

Other color spaces

File:
H&E.tiff



- Image/type/HSB stack or RGB stack
- Image/Adjust/Threshold Colour