MCDB/BCHM 4100/6440 – Microscopy Labs

Lab 1:

Diffraction-limited Imaging

Lecturer: Jian Wei Tay

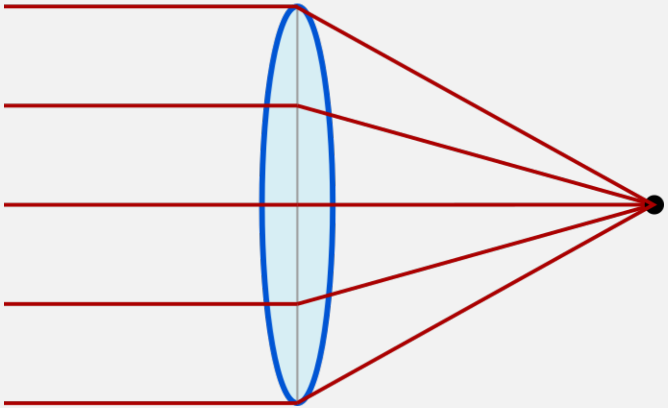
Date: 2021-09-13

Learning objectives

- What is a point spread function and how it affects imaging
- Finding objects in an image using intensity thresholding
- Measuring the size of a diffraction-limited object using curve-fitting

From Joe's lectures

- An image is generated when light is focused on a plane
- Focusing is caused by constructive interference of individual rays of light

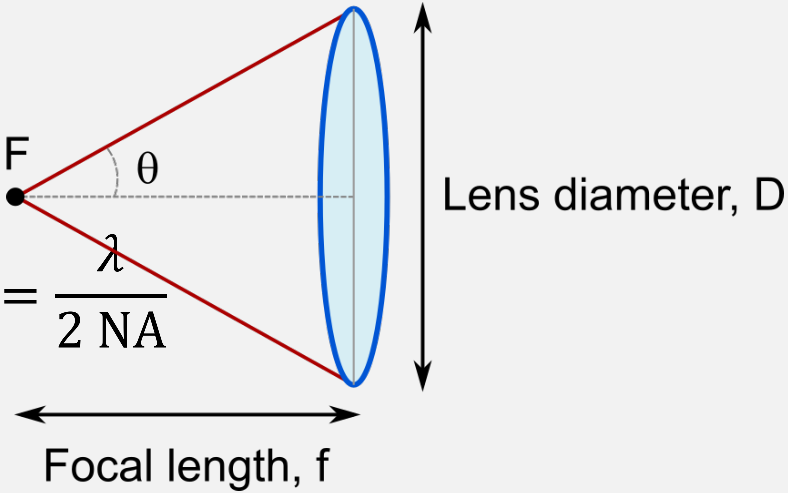


The diffraction limit

- The diffraction limit describes the smallest object that can be observed using a microscope

- The Abbe diffraction limit d is given by

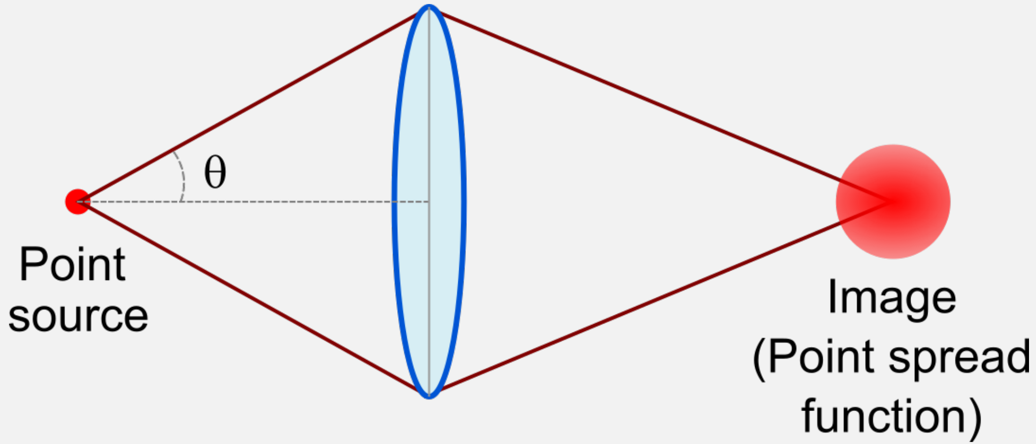
$$d = \frac{\lambda}{2n \sin\theta} = \frac{\lambda}{2\ \mathrm{NA}}$$



where θ is the acceptance half-angle

Diffraction-limit in imaging

- A point source (diameter << Abbe limit for given lens) is blurred due to diffraction
- The blurred image is called the point spread function



How the point spread function affects images

- The image generated by a lens is the original image convolved (blurred) by the point spread function

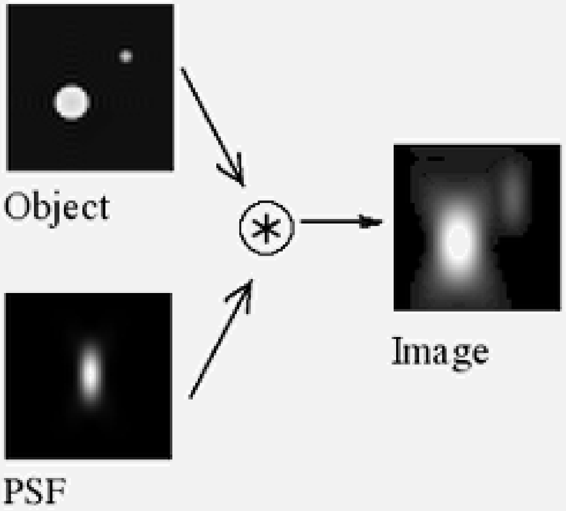
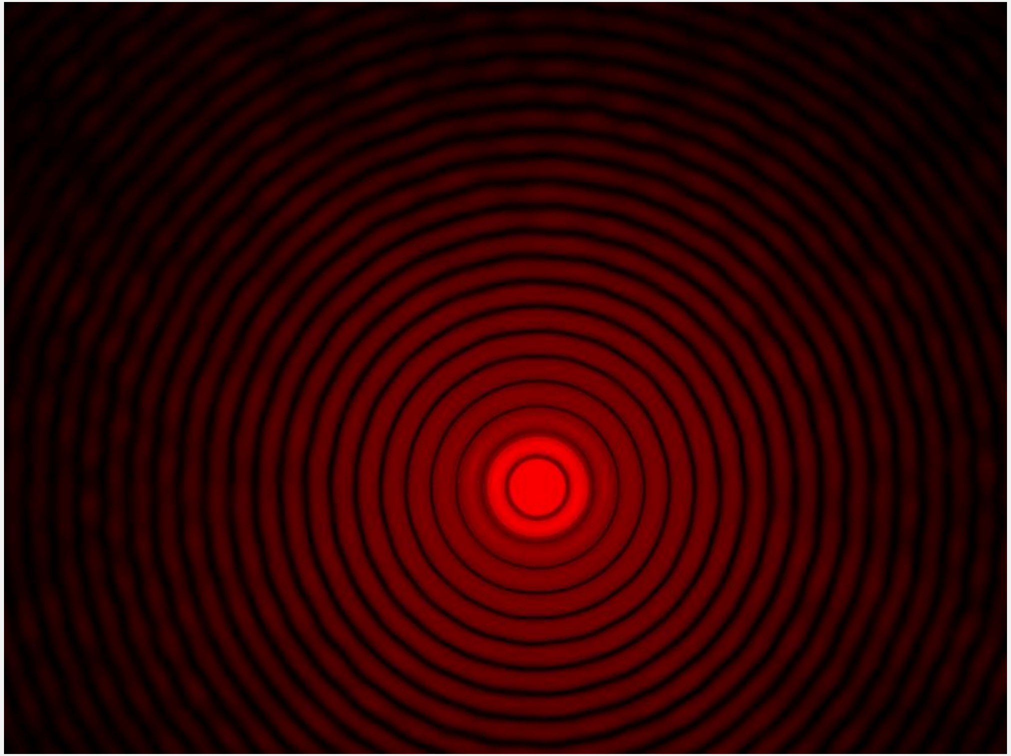


Image: [Wikipedia]

The point spread function of a perfect lens is the Airy disk



How the point spread function affects images

- The image generated by a lens is the original image <u>convolved</u> (blurred) by the point spread function

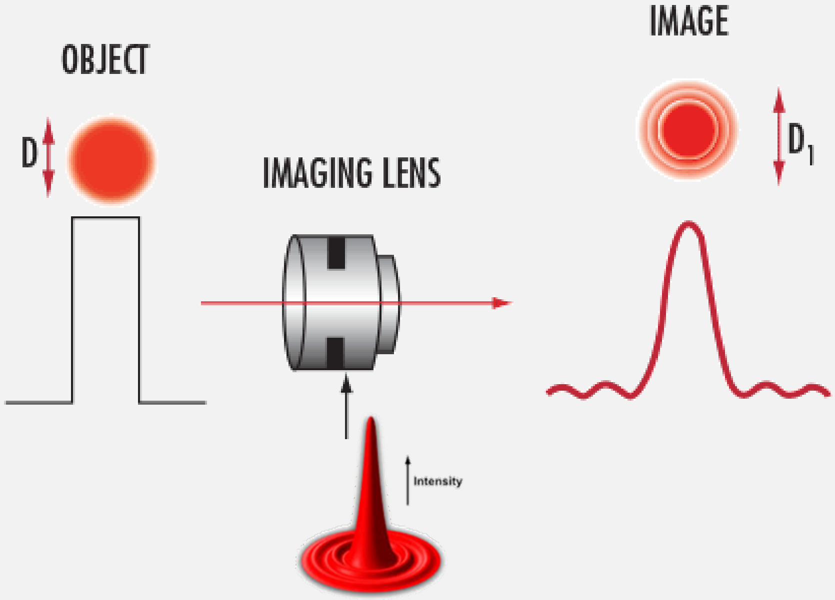
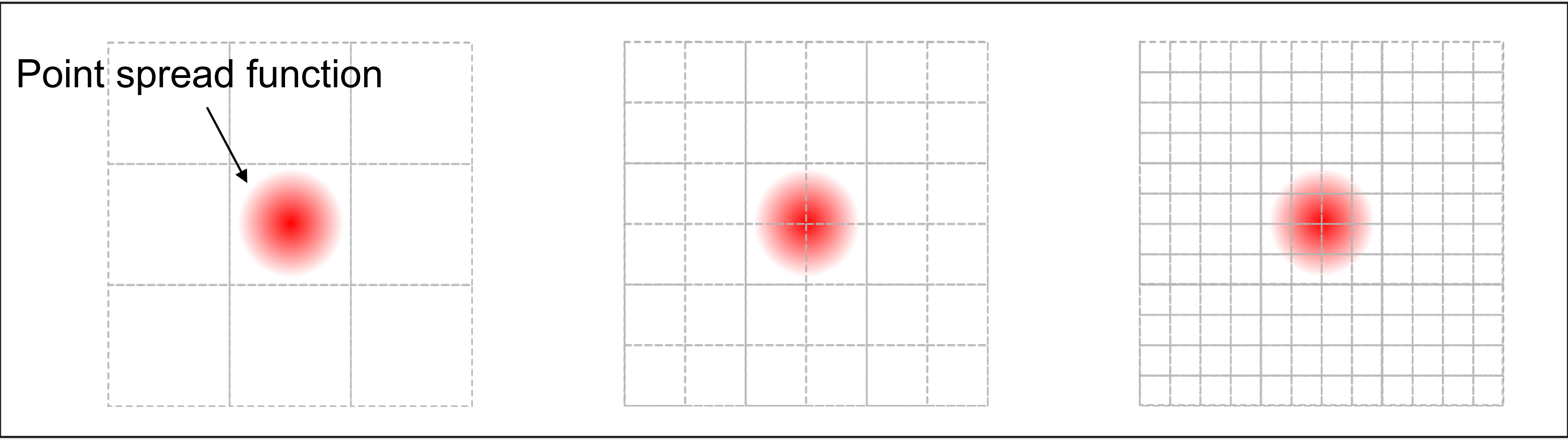


Image: [Edmund Optics]

Pixel size further limits imaging resolution



To accurately image an object, PSF must span at least 3 pixels (Nyquist limit). Only the setup on the right can accurately measure the object.

Questions?

One last note

- Lenses are good at focusing light in the lateral plane (perpendicular to direction of travel of light)
- Light focused along the direction of travel is not focused as well
- This means that z-resolution is much worse than x- and y-resolution (at least for some microscopes…)

How to measure the size of spheres in image

- To measure the size of a spherical object in an image, we can fit the object to the Gaussian equation

$$y = A \exp\left(-\frac{(x-B)^2}{2C^2}\right)$$

where A is the maximum intensity of the object, B is the x-offset, and c is the standard deviation.

How to measure the size of spheres in image

- The object size is approximated by the full-width at half-maximum (FWHM) of the fitted Gaussian

$$\mathrm{FWHM} = 2\sqrt{2\log_{10} 2}\; c$$

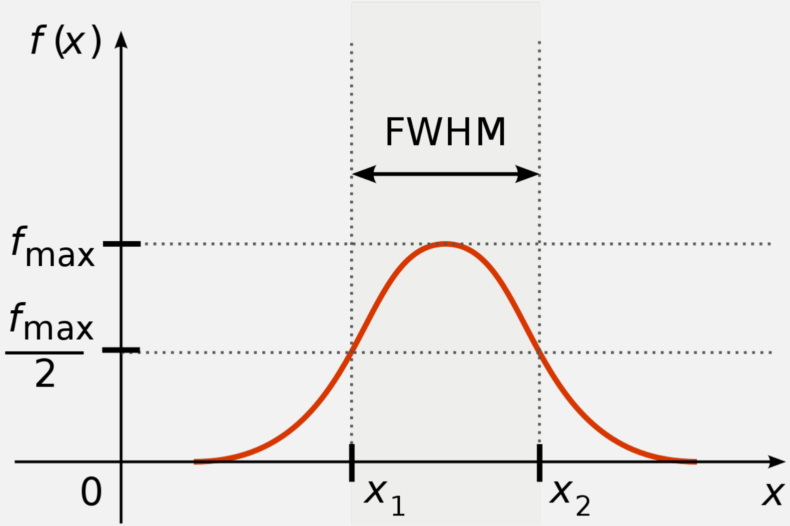
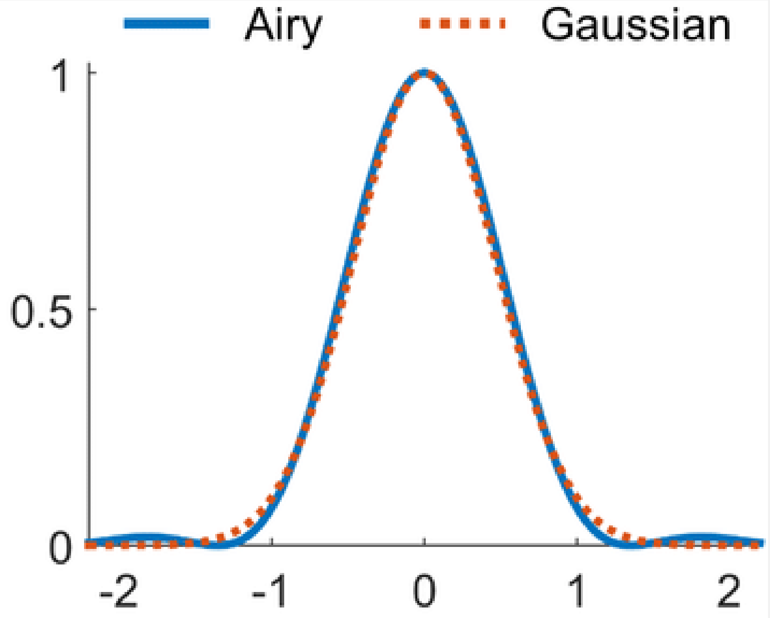


Image: [Wikipedia]

Why use a Gaussian?



- The Airy disk equation is computationally difficult to fit because it has Bessel functions which are complex
- The Gaussian equation is a good estimate of the central peak of the Airy disk (within a few %)

Image: ResearchGate

Questions?

Suggested image analysis protocol for Lab 1

- Threshold image to find bright beads
- Use `regionprops` to approximate the center of each spot
- Use subscript indexing to extract the intensity profile along either `x` or `y`
- Fit the resulting intensity profile to a Gaussian model

Intensity thresholding

- The basic idea is to create a mask (logical array) that is true where image intensity is above some threshold value
- The easiest way is to use the *greater than* operator >

Practice

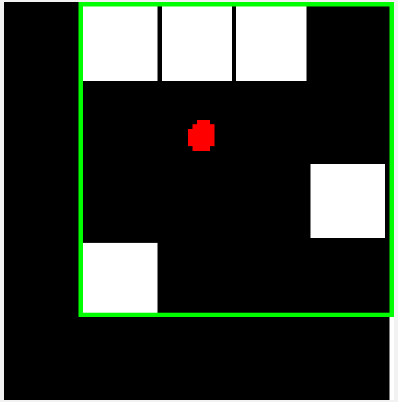
- Read in one of your non z-stack images
- Generate a mask of the beads

Use regionprops to measure the centroid

- Syntax:

```
stats = regionprops(mask, 'Centroid')
```

- The centroid is the center coordinate of the mask

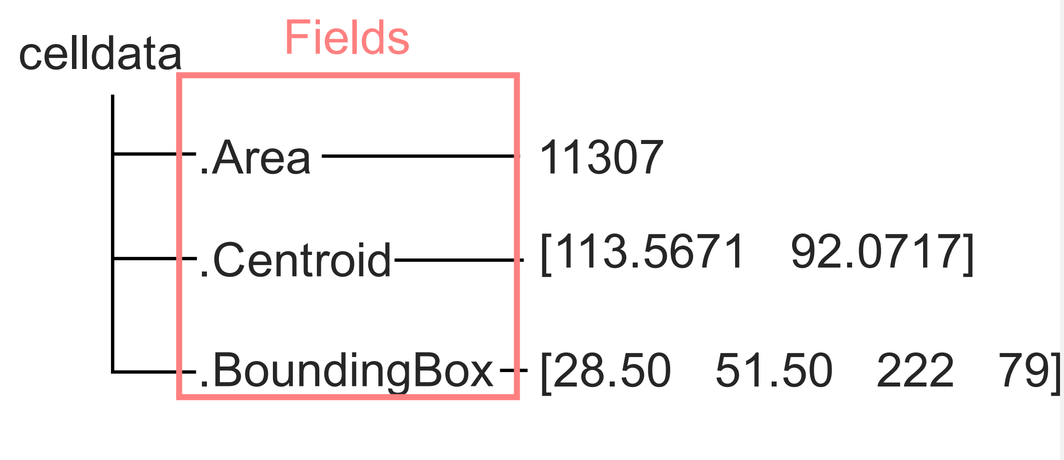


Note: Use the `help` function to learn how to use `regionprops`

```
>> help regionprops
```

Structured Arrays (`struct`)

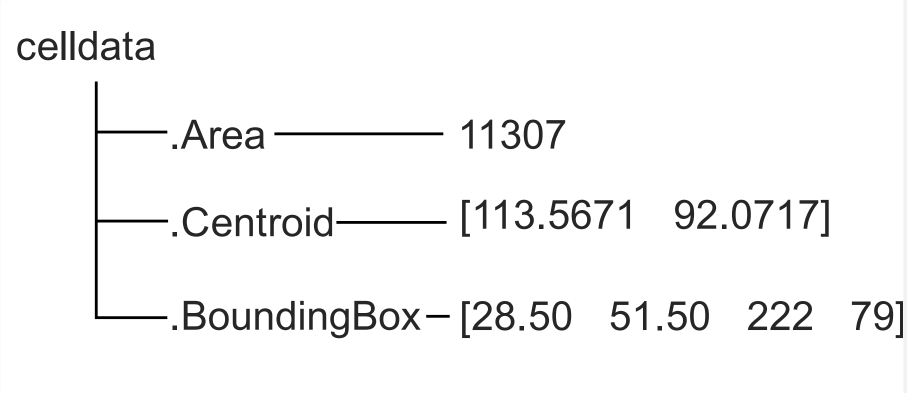
- `struct` is a basic MATLAB data type



- Data is stored in named **fields**

- Data stored in fields can have different data types and sizes

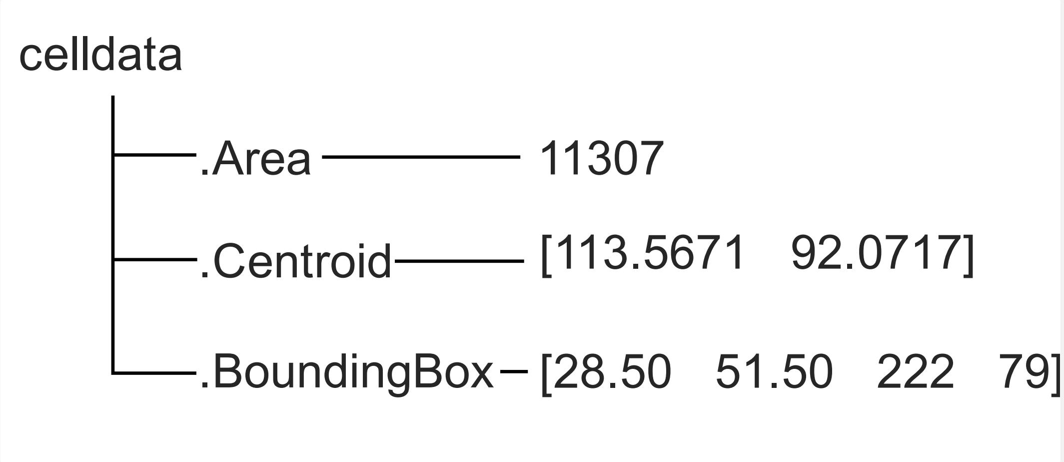
Accessing data from a struct



`>> celldata.Area` **(Dot notation)**

- Fieldnames are case-sensitive
- **For `regionprops`, output fieldnames always have uppercase first letters**

Number of detected objects = number of elements in struct




```
numCells = numel(celldata)
```

Collecting data from a struct into a matrix

- You can collect data from a struct into a single matrix using the `cat` function (short for 'concatenate' or 'join')

```
centroids = cat(1, celldata.Centroid);
X = centroids(:, 1);
Y = centroids(:, 2);
```

Note: I made a mistake in class. Turns out you can't put in the second index (e.g., cat(1, celldata.Centroid(1)) fails)

Collecting data from a struct into a matrix

- This was the alternative code from class using for loops

```
for idx = 1:numel(stats)
    X(idx) = stats(idx).Centroid(1);
    Y(idx) = stats(idx).Centroid(2);
end
```

Use subscript indexing to get the x or y intensity profile

- See notes from Lecture 4 (uploaded to lab course on Canvas)

Curve-fitting in MATLAB

- Use the function `fit`

```
fitobject = fit(x, y, fitType)
```

`fitType` should be a string that specifies the model to fit to

Note: Use the `doc` function to open the MATLAB documentation. Then search for “List of Library Models for Curve and Surface Fitting”.

Practice

- Add a line that uses the `fit` function to fit the intensity profile to a Gaussian model

Plot the fitted data

- To assess the goodness-of-fit, plot the data

```
plot(fitObj, x, y)
```

Getting the fitted parameters

- Use dot-indexing to get fitted parameters

```
A = fitObj.A
```

Adding an initial guess

```
curve = fit(x, y, fitType, 'StartPoint', p0 );
```

p0 = [A B C D]

Questions?

Repeating the fitting process using a for loop

- In MATLAB

```
for index = 1:nLoops

    %Statements to repeat

end
```

Example of looping through bead positions

```
mask = I > 2500;
stats = regionprops(mask, 'Centroid');

for index = 1:numel(stats)

    currCentroid = stats(index).Centroid;

    %Do fitting here


    %Store fitted bead diameters (in pixels)
    fittedDiameter(index) = fitObj.c1;

end
```

Questions?

Next week: Presentations

- The idea behind the presentations is to allow you to share and compare your results (since there are three different microscopes)
- It is also a place to ask questions or share interesting methods with the group

Presentation guidelines

- One person from each microscope group will present
- Treat it like presenting progress at group meeting
- Show us your results, images, and code
- You don't have to be completely done – you can ask questions and point out problems with code

Lab 1 report is due Sept 30 at the start of lab

- Each person needs to submit their own report via Canvas
- List the people in your group
- Explain your image analysis. **Write as though methods and results section of journal paper.**
- Provide representative images at appropriate steps (e.g., showing your mask). **Please NO LARGE IMAGES** - crop them so we see only a few objects.
- Include plots showing fitting and your results as appropriate. Label the axes of each plot.
- Include figure captions
- Include your full code at the end of the report

Images in lab reports

- Please crop your images