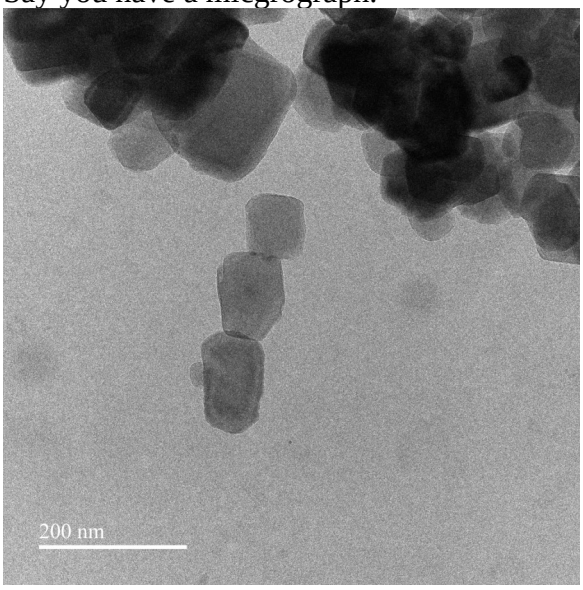
Manual for the Wagon Wheel Measurer

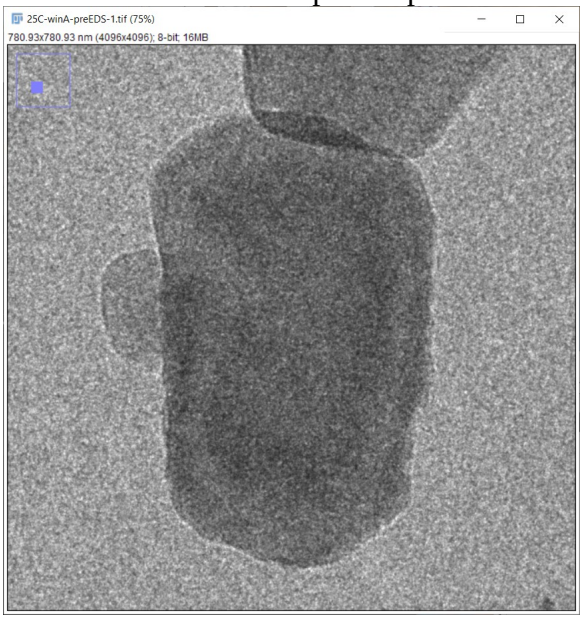
This program aims to take some of the subjectivity out of particle diameter measurement.

What does this program do

Say you have a micgrograph.

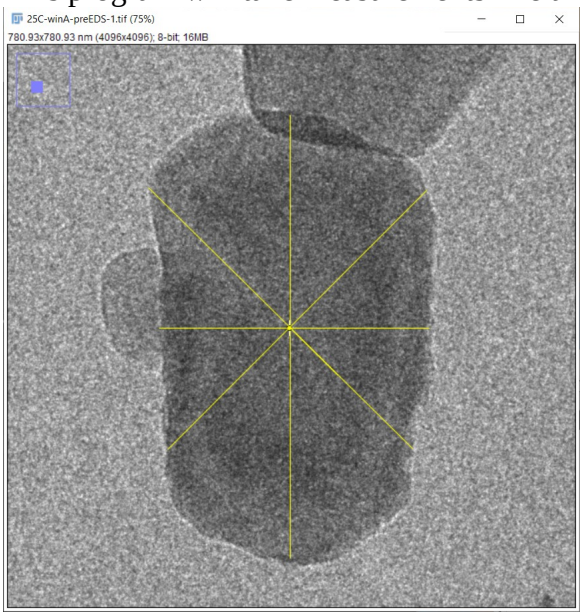


Lets take a look at this specific particle. What's its "diameter."



Well, that depends on how you measure it.

This program will take measurements like this:

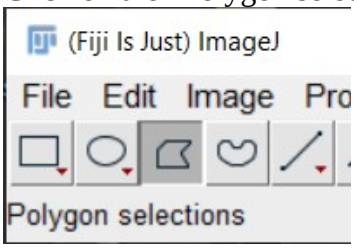


Each one passes through the particle's center of mass. They look a little bit like spokes on a wheel, thus the name of the program.

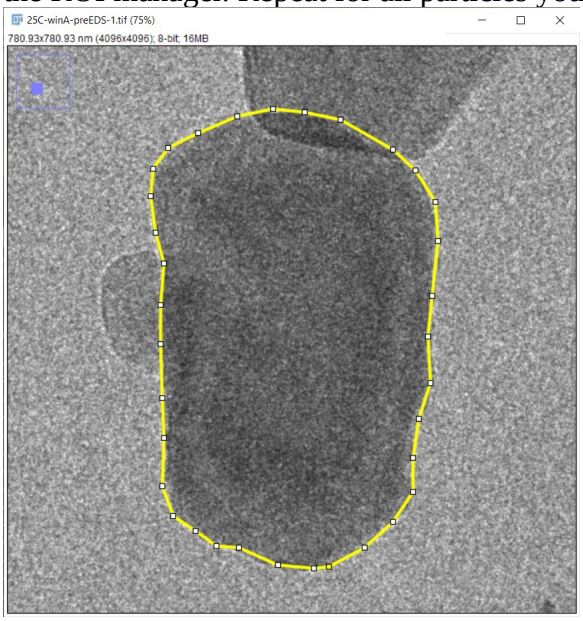
In the picture above, the program is taking 1 measurement every 45°, but you can make it take measurements every 20°, 10°, or even 1°. You'll get all the resulting diameters in a text file.
You can take the average to find the diameter if the particle was actually circular. You could also subtract the maximum diameter from the minimum diameter, and find out how ovoid your particles actually are. You could compare before and after micrographs to see if they change shape.

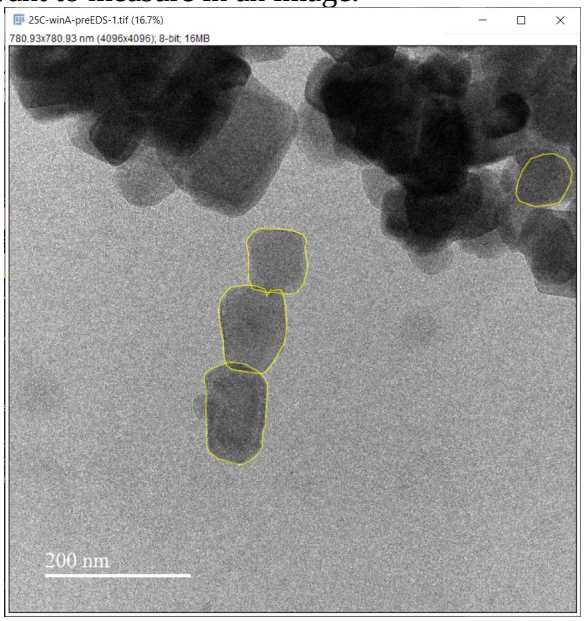
How do you use the program.

1) Measure your particles
   a) this program uses the ROI (region of interest) file format. You'll need to install ImageJ, or FIJI to use it
   b) Open your image
   c) Click on the "Polygon selections" tool

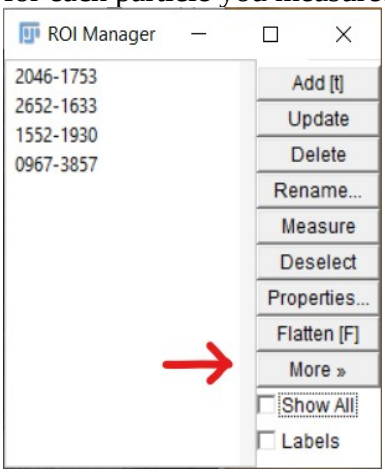


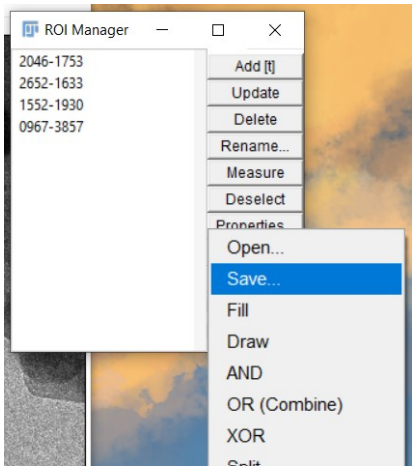
   d) Trace the outline of your particle. Press "t" once you're finished to store it (temporarily) in the ROI manager. Repeat for all particles you want to measure in an image.



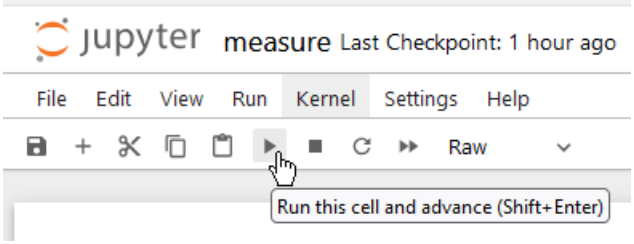


   e) In the ROI manager, click `More >>` then "Save" Use a filename you'll be able to match up to the image you measured from. The resulting file should be a `.zip` containing 1 `.roi` file for each particle you measured.





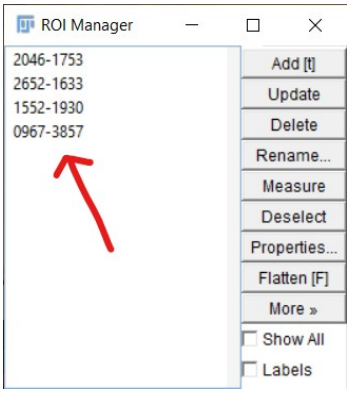
2) Open and run measure.ipynb
   a) This file is composed of "cells," the boxes with code or text in them. Look for the cell that starts with "1: Input the file paths and angles you want to measure." In order to run the program, you'll need to update the values in the cell below this.
      `angleStep` This is the angle between measurements. The example angleStep is 45°
      `zipFilePath` This is the path* to the `.zip` file you made in step 1
      `saveFilePath` This is the path* where you want to save your data
      *What's a path? Click here
   b) Select the first cell at the top. Click the "Run this cell and advance" button until you reach the bottom.



3) Open your output file
   a) even though it's a `.txt` file, you should be able to open the output file in excel, or a spreadsheet program of your choice. You'll end up with something like this.

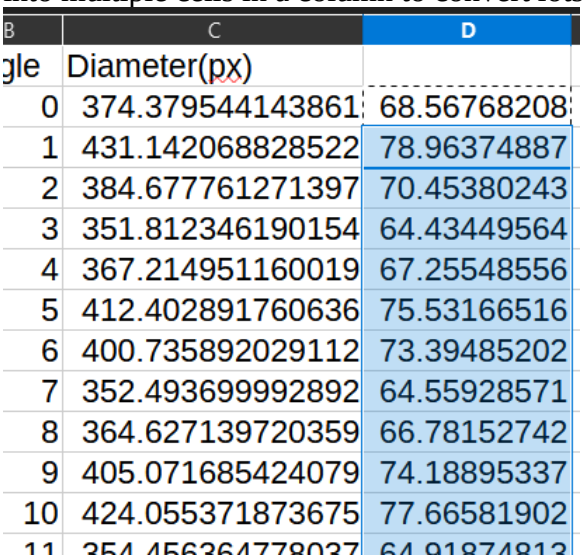
	A	B	C	D
1	ROI Name	Angle	Diameter(px)	
2	0967-3857.roi	0	374.379544143861	
3	0967-3857.roi	1	431.142068828522	
4	0967-3857.roi	2	384.677761271397	
5	0967-3857.roi	3	351.812346190154	
6	0967-3857.roi	4	367.214951160019	
7	0967-3857.roi	5	412.402891760636	
8	0967-3857.roi	6	400.735892029112	
9	0967-3857.roi	7	352.493699992892	
10	0967-3857.roi	8	364.627139720359	
11	0967-3857.roi	9	405.071685424079	
12	0967-3857.roi	10	424.055371873675	
13	0967-3857.roi	11	354.456364778037	
14	0967-3857.roi	12	368.807050462138	
15	0967-3857.roi	13	402.339801692424	
16	0967-3857.roi	14	432.657266092189	
17	0967-3857.roi	15	359.053033038949	
18	0967-3857.roi	16	372.210157524229	

The first column are the names from the ROI manager in step 1e.



The second column is the angle the measurement is taken at
The third column are your actual diameters

b) To convert the diameters the program gave you from pixels to real-world distance, check the scale bar, and use a formula to multiply them. For example, if your image had a 273 pixel wide scale bar, labeled “50 nm” you’d use the formula `=C2*(50/273)` Where the value at position C2 on your spreadsheet is a diameter in pixels. You can copy/paste your formula into multiple cells in a column to convert lots of measurements at once.



B	C	D
gle	Diameter(px)	
0	374.379544143861	68.56768208
1	431.142068828522	78.96374887
2	384.677761271397	70.45380243
3	351.812346190154	64.43449564
4	367.214951160019	67.25548556
5	412.402891760636	75.53166516
6	400.735892029112	73.39485202
7	352.493699992892	64.55928571
8	364.627139720359	66.78152742
9	405.071685424079	74.18895337
10	424.055371873675	77.66581902
11	354.456364778037	64.91874813

c) Hopefully this is enough to get you started. I’ll add more to this manual as I continue updating the program.

File paths

A file path is an easy way to tell a program where you put a file. For example, say you're storing your images for research in a folder on your desktop. The path to one of those photos might be
Windows: `C:\Users\[your username]\Desktop\microscopeProject\image.tif`
Mac/Linux: `/Users/[your username]/Desktop/microscopeProject/image.tif`

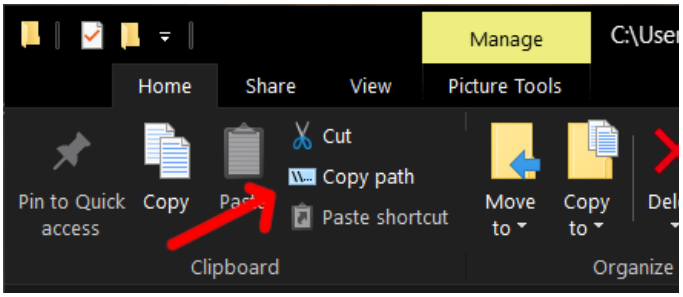
Every OS puts the option to copy a file's path somewhere different. I've included screenshots, but if they're out of date, just look up "find file path" and then the name of your os.

To put the file path into the python program, you need to put quote marks around it. Always put an `r` before your opening quote mark. Windows file paths have backslashes (`\`) which mess with the program. The `r` tells the program to ignore them though.

One final note. If you use the same output file, and run the program multiple times, all of the results will end up in that one file, one after another. The code just appends new measurements at the bottom.

In the end, the lines should look something like this
Windows:
```
zipFilePath = r"C:\Users\username\Desktop\projectFolder\zipFile.zip"
saveFilePath = r"C:\Users\username\Desktop\projectFolder\output.txt"
```
Mac/Linux:
```
zipFilePath = r"/Users/username/Desktop/projectFolder/zipFile.zip"
saveFilePath = r"/Users/username/Desktop/projectFolder/output.txt"
```

Windows 10



MacOS Sequoia