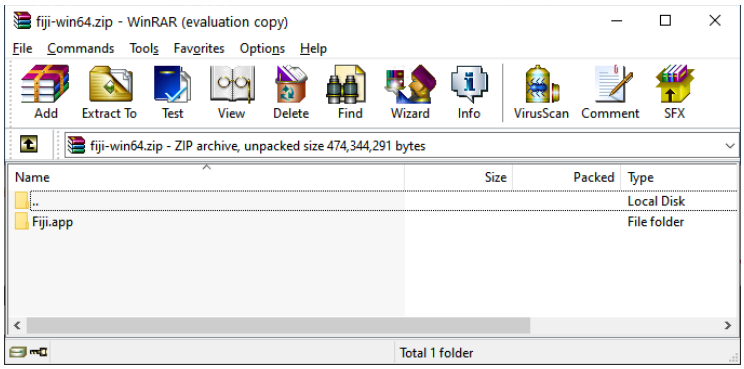
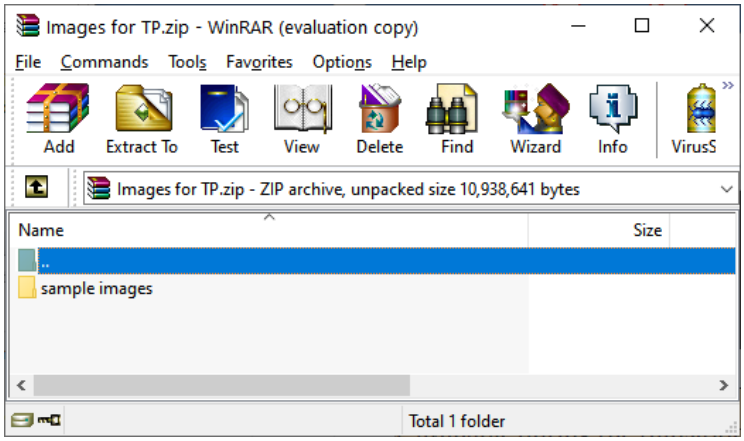
Download Fiji https://imagej.net/Fiji/Downloads

Place it on desktop and unzip (drag Fiji.app on desktop)

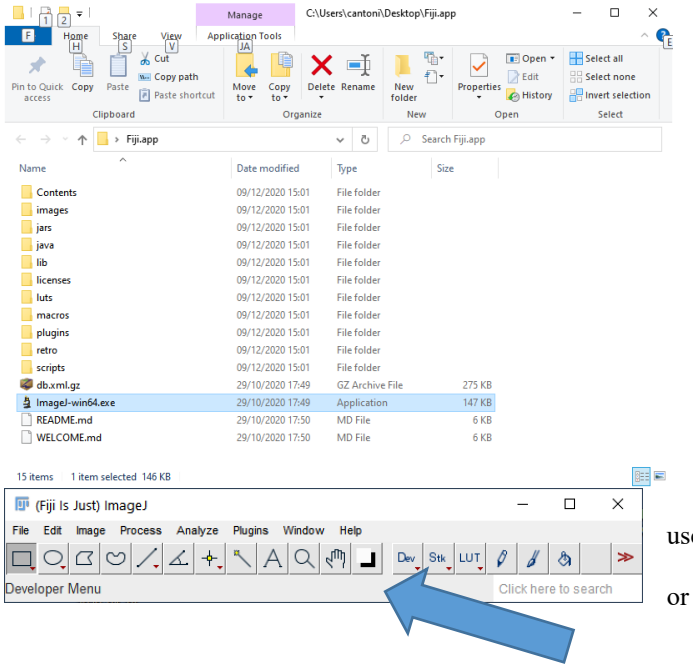



Download and extract sample images from moodle




---

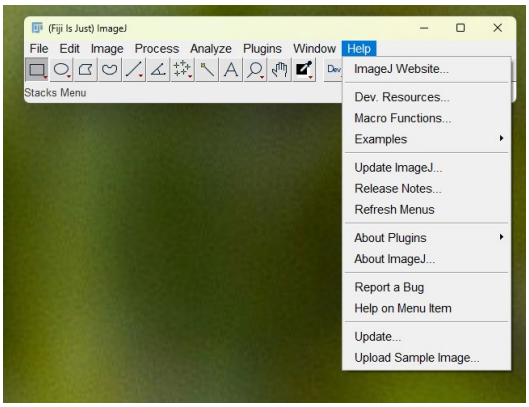
## 1. Start the program:




use File -> open command

or drag & drop

## 2. Getting help:



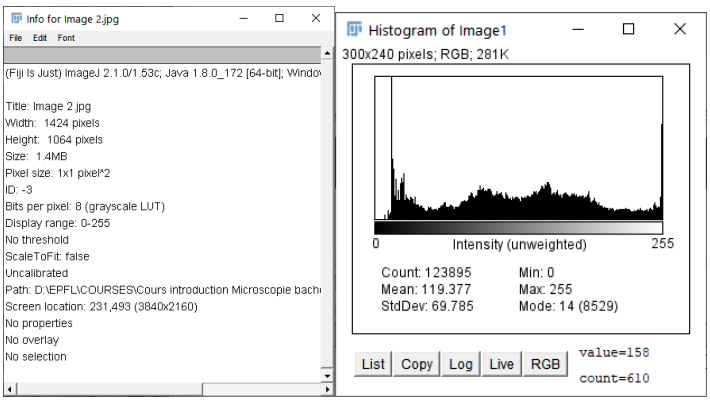

Documentation_ (nih.gov)
Getting help: ImageJ User Guide - IJ 1.46r (nih.gov)

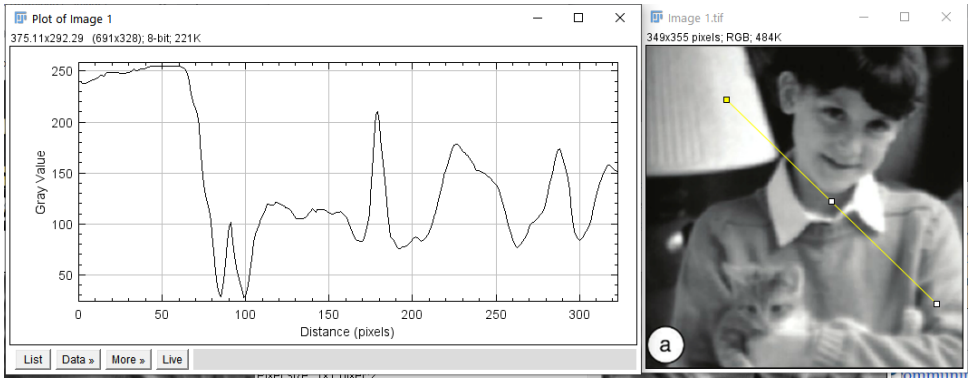
## 3. Preparing the data

The quality of the analysis depends very much on the quality of the data!

1. Importing images: Drag & drop or File → Open Image 1, Image 2 Image 3
   NOTE information on image window

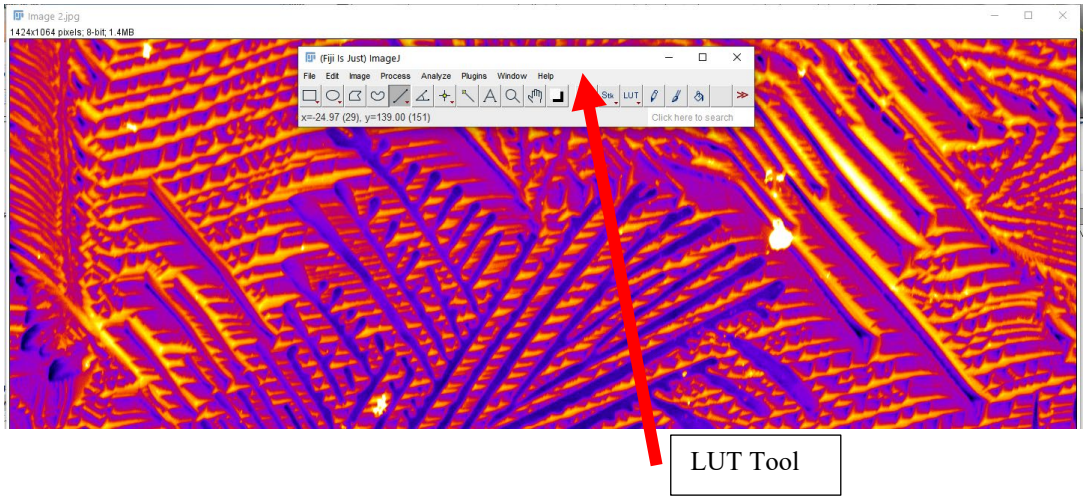
2. Exploring the image:
   a. Image → Show info
   b. Analyze → Histogram || of whole image and of a selection of the image
   c. Analyze → Plot Profile: this requires a *selection*



3. Scale & Calibrate (Image 2 and Image 3)
    a. Select full length of the scalebar with *line* tool
    b. Analyze → Set Scale

4. Using color
    a. Image → Color → Show LUT
    b. Image → Color → Display LUTs
    c. Image → Lookup Tables → Select a table
    d. ALSO LUT tool & Lookup Tables Tool




## 4. Image Processing

Preparing the image for subsequent analysis

Open Image 1 and make copies (Image -> Duplicate)
add some noise:
a) Process -> Noise -> Salt and Pepper
b) Process -> Noise -> add Noise

1. Using Filters
    a. Process → Filters → Convolve (See the matrix)
    b. Process → Filters → Gaussian Blur (choose sigma, 2&15)
    c. Process → Filters → Median (choose radius, 2&15)
    d. Process → Filters → Variance (choose radius, 2&15)
    e. Compare Gaussian Blur, Median, & Variance
    f. Try different matrix filters:

**3x3 average**

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

**5x5 average**

| | | | | |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

**gaussian**

| | | |
|---|---|---|
| 1 | 2 | 1 |
| 2 | 4 | 2 |
| 1 | 2 | 1 |

## 5. Spatial Domain Image Enhancement

Knowing what the options represent (Image 1, Image 2, Image 3)

1. **Contrast Manipulation**
    a. Image → Adjust → Brightness/Contrast (Linear)
    b. Check Histogram: Analyze → Histogram
    c. Process → Math → Gamma (Non-Linear), check Histogram before and after

2. **Contrast Enhancement**
    a. Process → Enhance Contrast ⇒ enhances image contrast by using either **histogram stretching** (grey levels) or **histogram equalization** (using the sum of pixel grey levels values as the transfer function)
        i. Saturated pixels value: number of pixel values that are allowed to become saturated
        ii. **Normalize:** it recalculates the pixel values of the image so that the range is equal to the maximum range of the data type
        iii. **Equalize Histogram:** takes the square root of the histogram values to equalize the contrast based on a selection

3. **Sharpening**
    a. Process → Sharpen ⇒ Invokes a Laplacian filter: accentuates detail in the image but can also increase noise
    b. Process → Filters → Unsharp Mask: subtracts a blurred copy of the image
        i. Radius: the standard deviation of the Gaussian blur
        ii. Mask Weight: determines the strength of filtering (=1 would be an infinite weight of the high-pass filtered image that is added
    c. Try to do a manual unsharp mask:
        i. Duplicate image and use gaussian blur
        ii. Use Process → Image Calculator… to subract blurred image from original image and add the result to the original image.

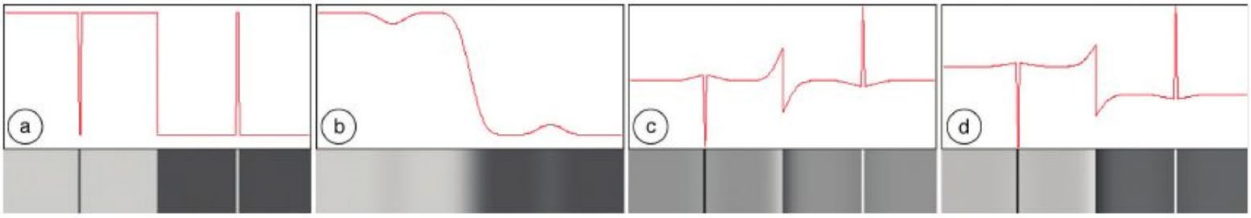



***Figure 5.28*** *Operation of an unsharp mask, with line profiles of brightness: (a) original image; (b) Gaussian smoothed; (c) subtracting the smoothed copy from original; (d) adding the difference (c) to original.*

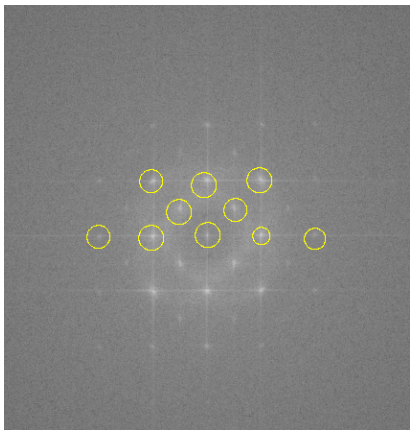
    d. Process → Find Edges ⇒ Invokes two 3x3 Sobel Filters to generate vertical and horizontal derivatives
    e. Use Process → Filters → Convolve… to create your own sobel Filter

| (Robinson) | (Sobel) | (Prewitt) | (Kirsch) |
|---|---|---|---|
| $\begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix}$ | $\begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}$ | $\begin{bmatrix} +1 & -1 & -1 \\ +1 & +2 & -1 \\ +1 & -1 & -1 \end{bmatrix}$ | $\begin{bmatrix} +5 & -3 & -3 \\ +5 & 0 & -3 \\ +5 & -3 & -3 \end{bmatrix}$ |
| $\begin{bmatrix} +1 & +1 & 0 \\ +1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}$ | $\begin{bmatrix} +2 & +1 & 0 \\ +1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}$ | $\begin{bmatrix} +1 & +1 & -1 \\ +1 & +2 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ | $\begin{bmatrix} +5 & +5 & -3 \\ +5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$ |

## 6. Image Enhancement in the Frequency Domain, using FFT

The aim is to find the defects in a transmission electron micrograph of $CoFe_2O_4$ with FFT manipulation

1. Open Image 5
2. Set correct scale in nm
3. Crop to make the image square
4. Save image as .tif
5. Process → FFT → FFT
6. Set correct scale: Image → Properties set nm to 1/nm



**Fourier Filtering** (direct way)

7. Make a circle on major reflections (use ctr-shift selection)
8. Edit → Fill (make sure that foreground color is white)
9. Process → FFT → inverse FFT
10. Do FFT of filtered image
    (due to inversion symmetry in FFT selection on only half a "hemisphere" is necessary)

**Fourier Filtering using mask**s, understanding the structure and application of masks

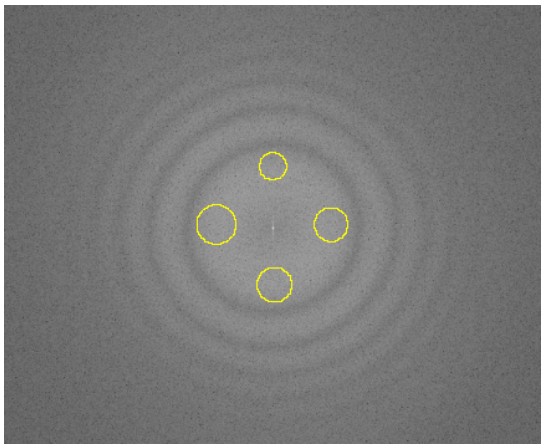
11. Apply a mask: Remember: Black areas (pixel value = 0) cause the corresponding frequencies to be filtered (removed) and white areas (pixel value = 255) cause the corresponding frequencies to be passed
12. On FFT, Make selections on vertical dots and fill them in (command F or Edit → Fill)
13. On FFT, Image → Adjust → Threshold
14. On FFT, Process → Binary → Convert to Mask, save as FFTmask.tiff
15. On image, FFT → Custom Filter (select FFTmask.tiff as filter)

Explore images 7 and 8

16. Look at their FFT
17. Filter images by selecting certain fourier frequencies in order to highlight different lattice planes (use inverse FFT)

Explore images 9 (amorphous carbon film)

18. Select a filter that would make sense for a periodic structure (like below)

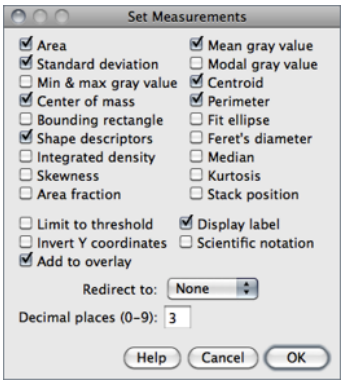


19. Observe what you filter did to image !

## 7. Segmentation & Thresholding, Particle Analysis and Image Measurements

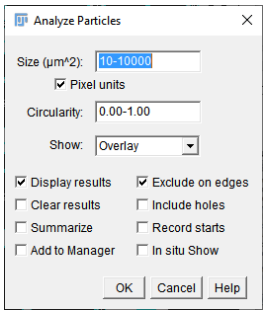
The aim is to get the size distribution of the grains in Image 10.

1. Open Image 10
2. Set correct scale in µm or nm
3. Crop to remove the operating parameter bar
4. Save image as .tif
5. Process → Filters → Gaussian Blur (sigma 2)
6. Process → Filters → Unsharp Mask (sigma 2, mask weight 0.9)
7. Image → Adjust → Threshold (66-255)
8. Process → Binary → Watershed (to see exactly what it does activate Edit → Options → Misc and activate Debug mode)
9. Save binary image
10. Analyze → Set measurements

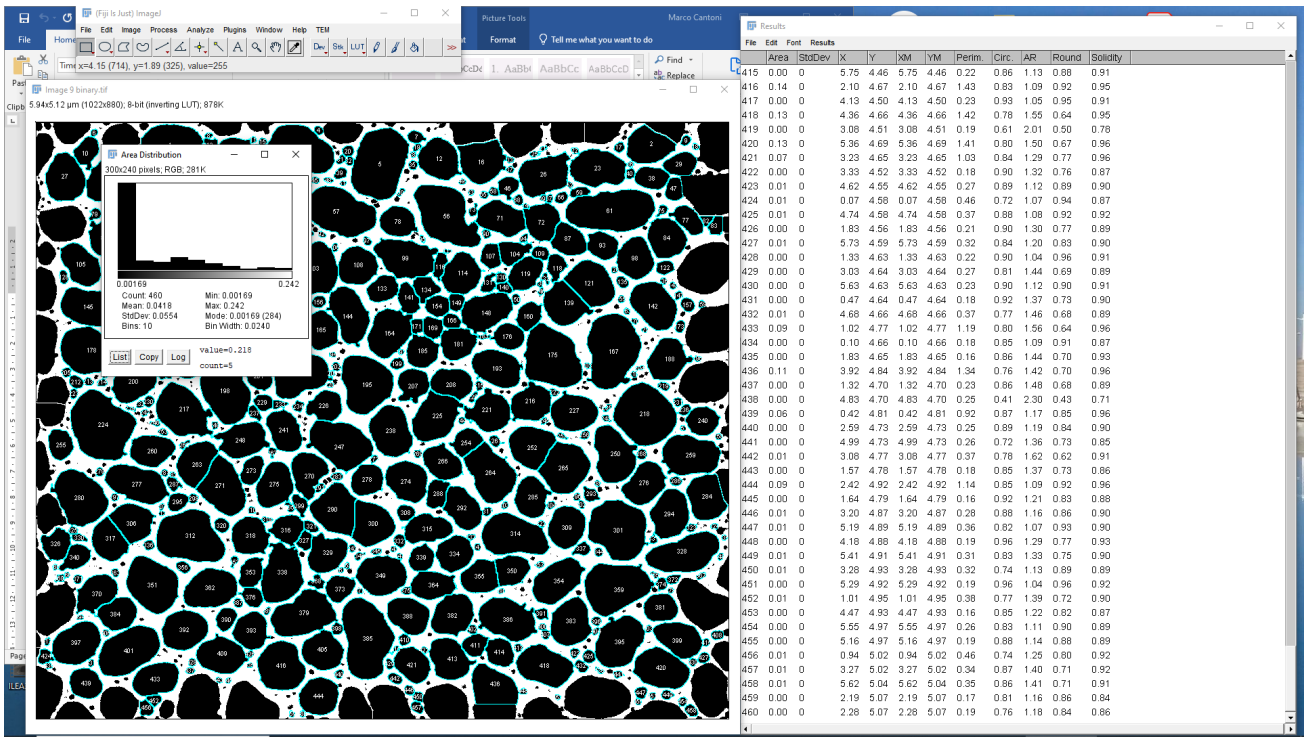



11. Analyze Particles

    a. Size: define the range of the particle sizes for the measurements (try 0.10 to infinity & then 0 to 0.10)
    b. Circularity: ranges from 0 (infinitely elongated polygon) to 1 (perfect circle) – leave as is
    c. Show: what to depict after calculation
    d. Exclude on edges: do not take into account particles at the edge of the image
    e. Check help



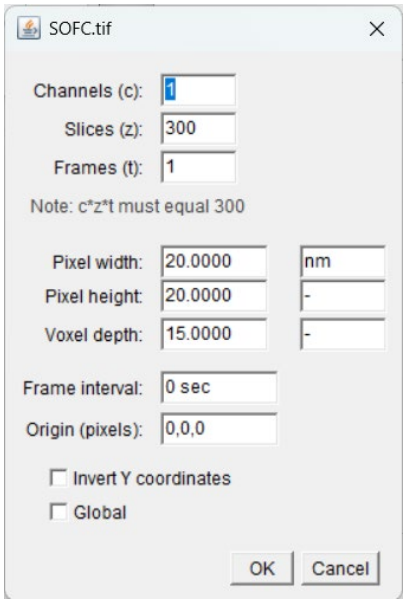

12. Analyze → Distribution



## 9. 3D Volume

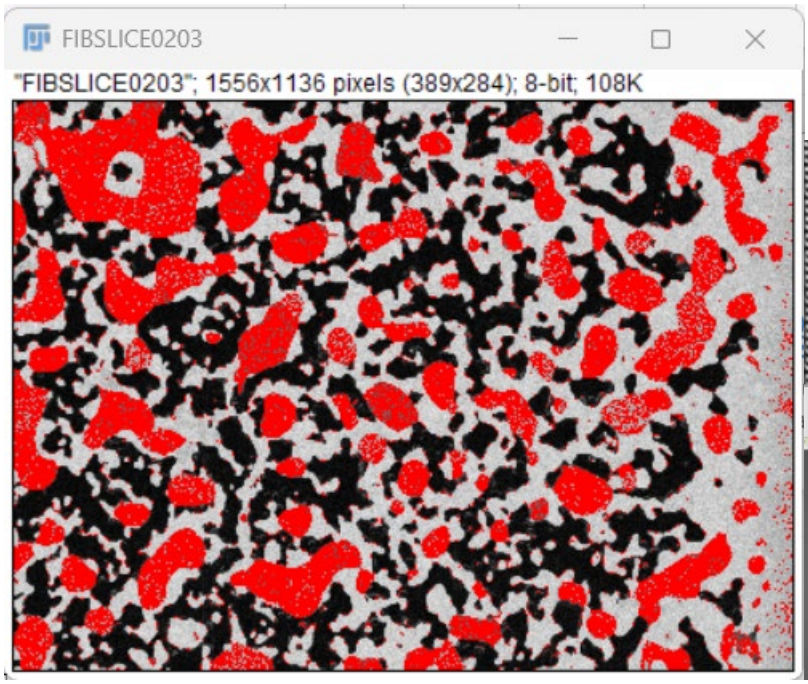
Visualization of 3D stack of images

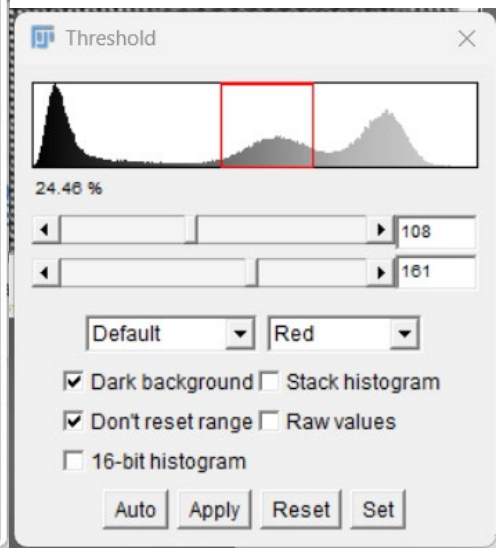
- Open SOFC.tif
- Image -> Properties
  adjust dimensions to 20/20/15nm voxel size
- Visualize orthogonal views
  Image -> Stacks -> Orthogonal Views
- Plugins -> 3D Viewer
- Play with Edit -> change Transparency or Edit Threshold



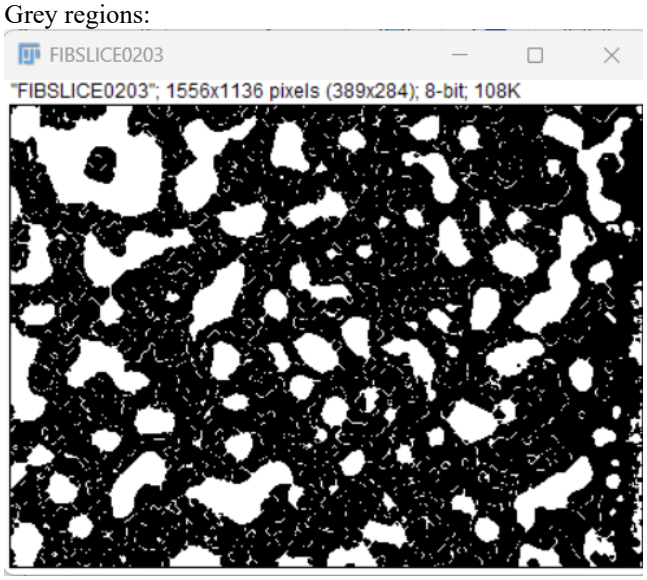

Segmentation in 2D

- Duplicate 1 image
- Process -> Filter _> Median
- Image -> Adjust -> Threshold…
- Make 3 segmented images for white, grey and black regions



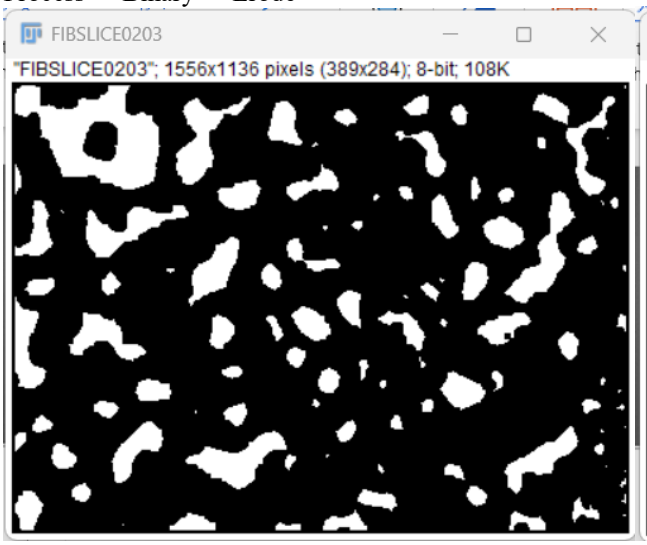




Grey regions:

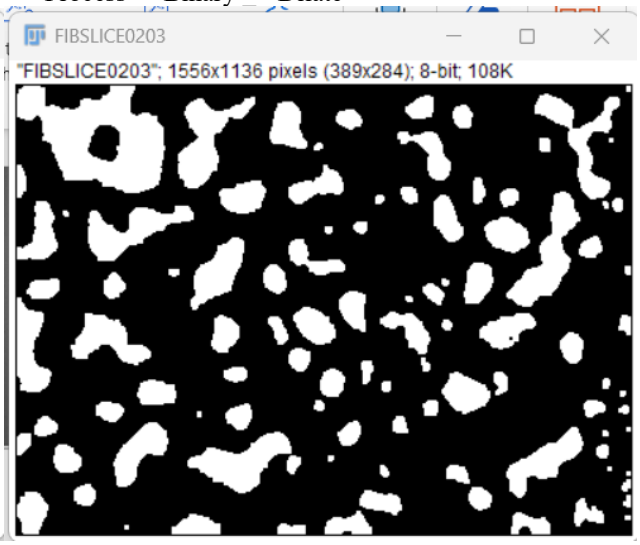



remove filaments !

Process -> Binary -> Erode

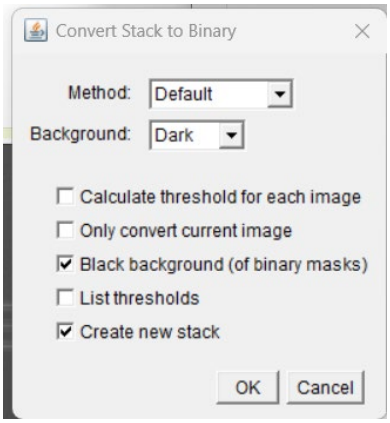


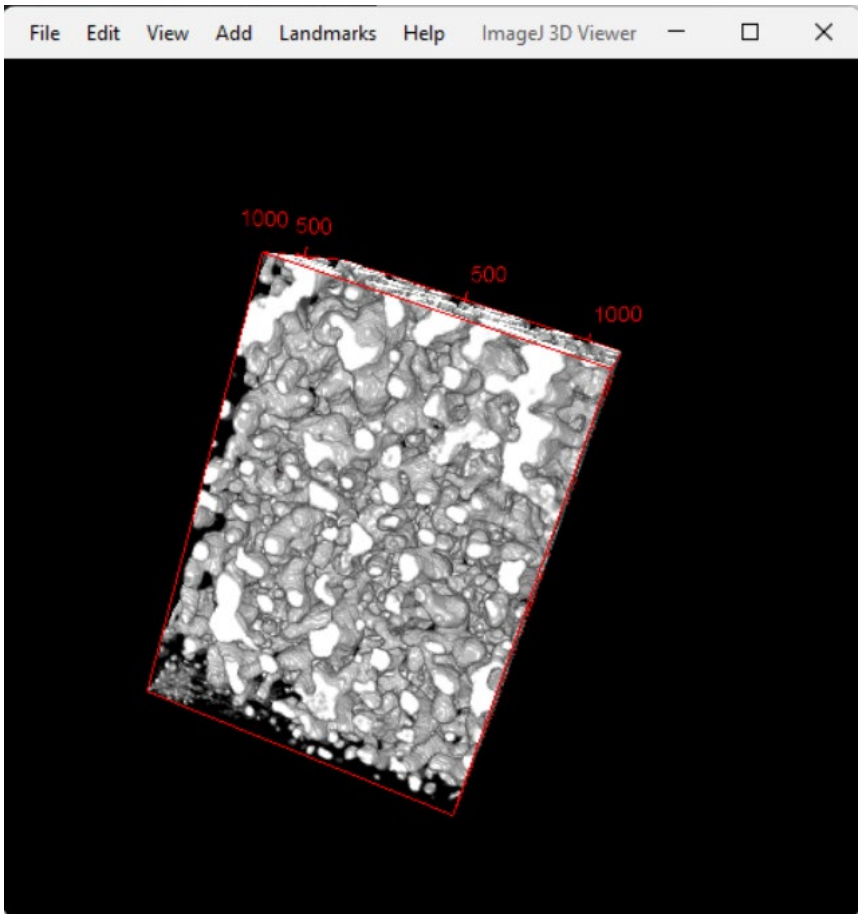

Process -> Binary _> Dilate



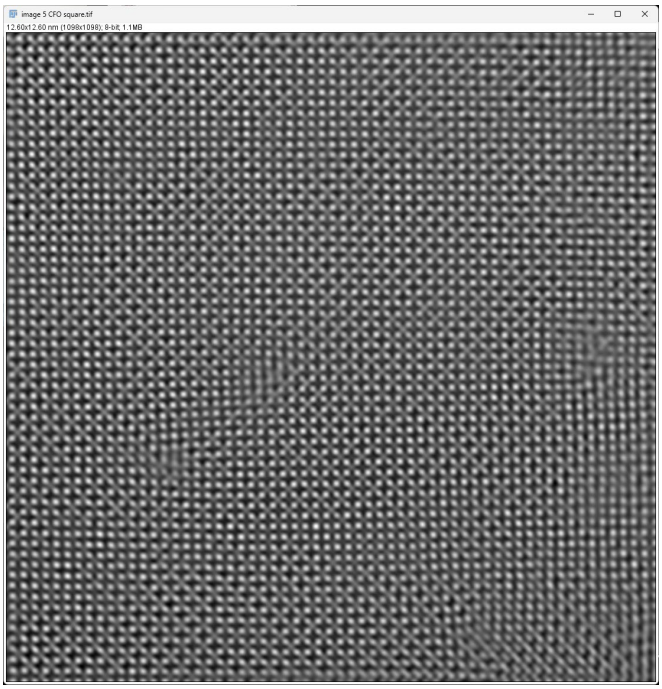

**In 3D:**

- Image Adjust Threshold (Stack)
- Erode & Dilate
- 3D Viewer

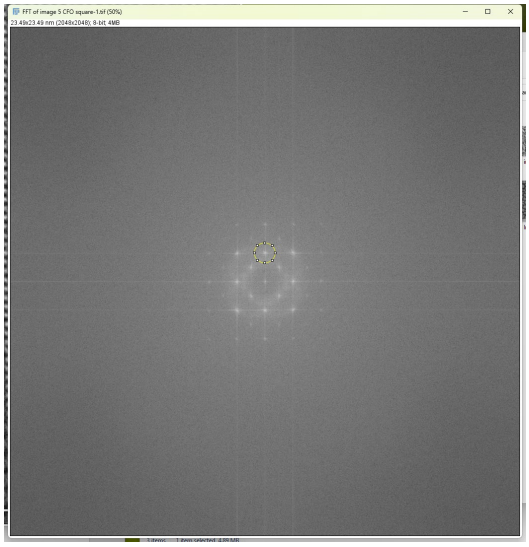


## 10. Example of Fourier filtering



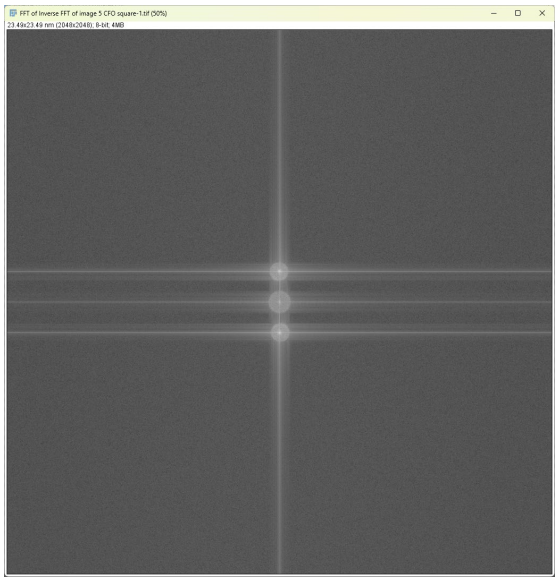
Image



FFT

FFT Filter (white = pass)

Inverse FFT (with filter)



FFT of filtered image