**Tools**

**Fiji (is just Image)**

* + ImageJ software but with all the useful plugins already installed
  + good for figuring out a pipeline that works
    - faster than implementing same pipeline in code

**Python(libariies)**

1. OpenCV
   * Faster than PIL
   * preferred library when doing image processing
2. Pillow (PIL)
   * Usable with Python's GUI tkinter and customtkinter
3. scikit modules
   * **ndimage**
   * **scipy.signal**
   * Has various submodules that are useful for more complex image processing
   * has some of the functions implemented that are not implemented in OpenCV
     + exp: probabilistic Hough transform
4. Shapely
   * Useful for finding intersections
5. REMBG
   * Neural network for removing backgrounds in Python

[**Image.sc**](http://Image.sc)

* Discussion forum for image processing

**Resources:**

*Image Processing*

Prof’s old website: <https://courses.cs.washington.edu/courses/csep576/20sp/>

This channel is very good to get a succinct overview of the techniques listed below:

<https://www.youtube.com/@firstprinciplesofcomputerv3258>

*Python + OpenCV*: <https://www.youtube.com/watch?v=oXlwWbU8l2o&ab_channel=freeCodeCamp.org>

**Image Manipulation**

**Morphological Operations**

<https://docs.opencv.org/3.4/db/df6/tutorial_erosion_dilatation.html>

Inversion

Thresh holding

Erosion

Dilation

Opening

Closing

**Noise Removal**

1. Median blur
2. Bilateral blur
3. (sometimes) Gaussian Blur
4. Bilateral filter (edge persevering while removing noise)

* These are all filters!

**Edge detection**

1. Canny Edge Detection
2. Binary threshold
3. Watershed algorithm