27555 Research
Progress Report 2
Feb. 6, 2017

Elim Zhang
yilinz@andrew

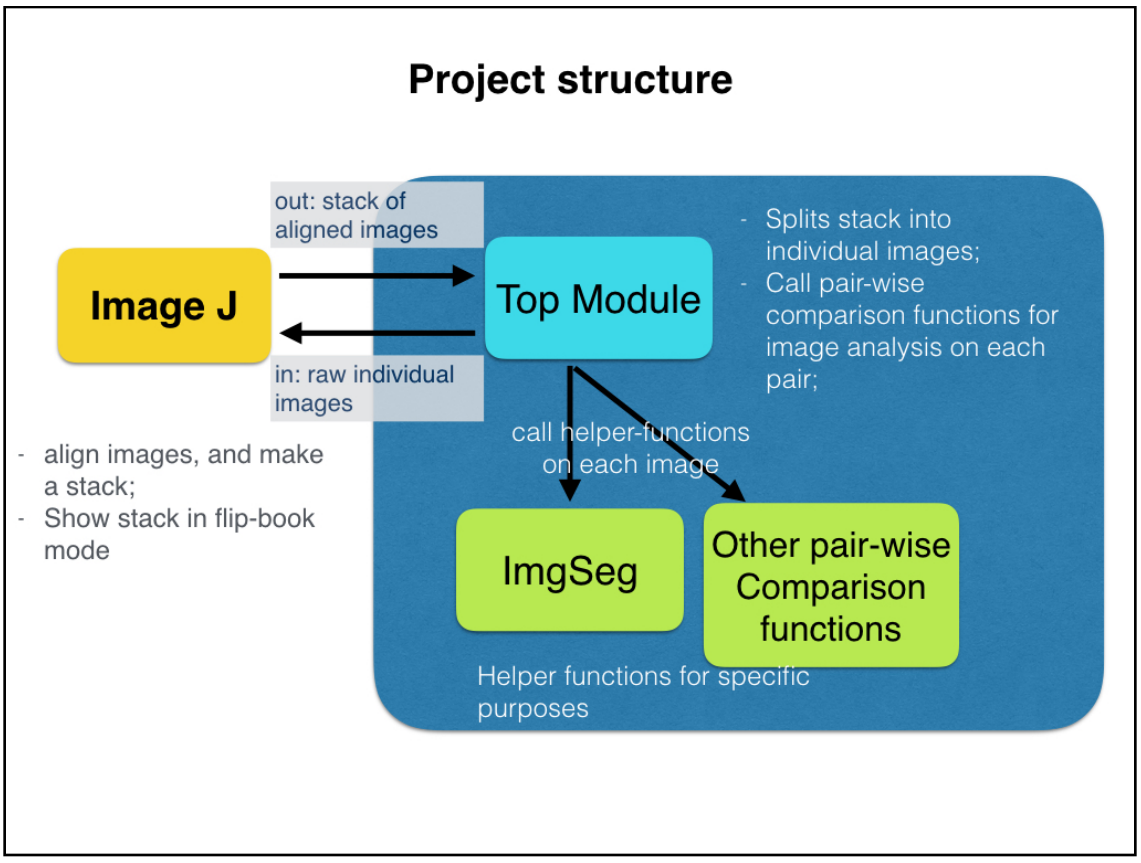
Progress Report 2
Feb. 6, 2017

Goal:

- **New project structure**; more modular (diagram below)
  - Top module: reads in a larger data set; and call helper function for image manipulation on each pair / individual images;
  - Helper functions pair-wise comparison functions
    - image segmentation on each image;
    - pair-wise comparison to refine segmentation results;

- **Version Control:** Set up a Git Repository for Version Control and easier take-over or code sharing;
  - Could easy go back to earlier versions of code;

- **Image alignment** using ImageJ;
- **Image Segmention:**
  - Refine result: reduce noises by comparing adjacent frames,
  - Perhaps track key features between frames;



27555 Research
Progress Report 2
Feb. 6, 2017

Elim Zhang
yilinz@andrew

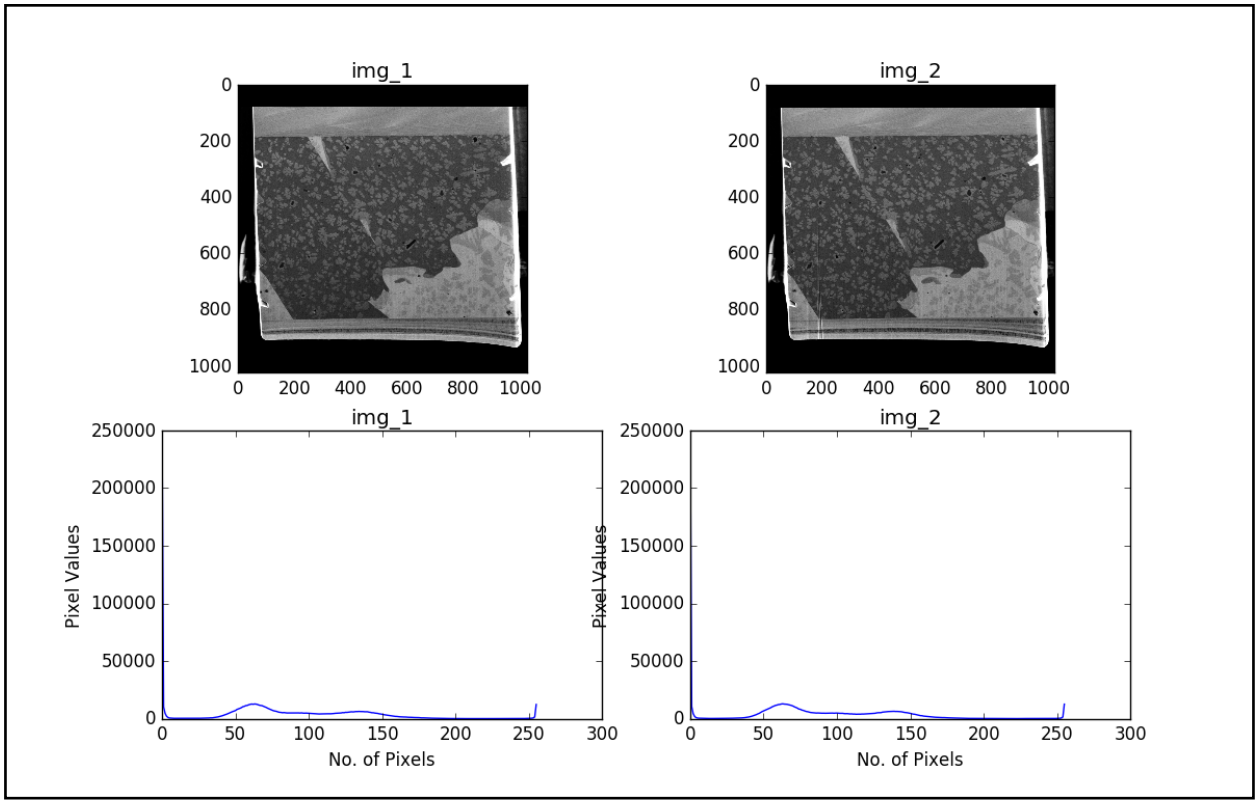
Trials and Thoughts:

1. For align slices in stack:

1) ImageJ: When choosing bounding rectangles, should first go to the center image, and set the bounding rectangle there;
2) Attempts: Draw bounding rectangle at different location of the frame;
   Conclusion: Setting bounding rectangle to the bottom edge of image works better, compared to setting it to any features; drastically improved compared to setting the box at top
3) Still slightly shaky after aligning;

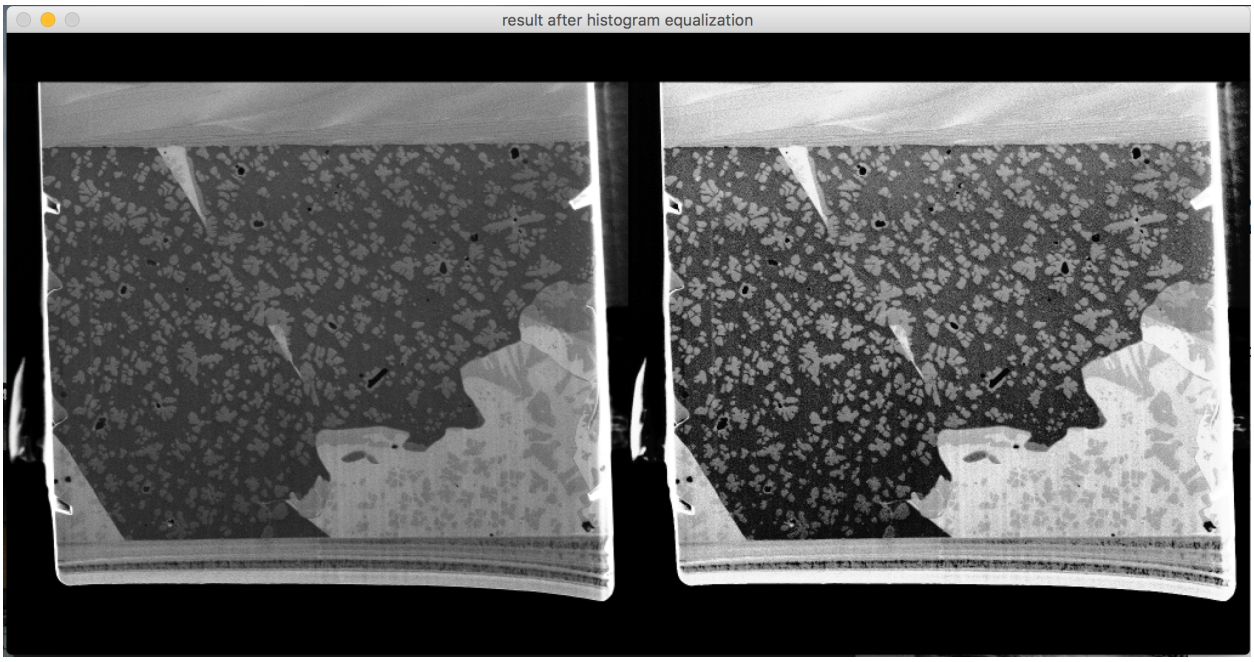
2. Histogram Equalization

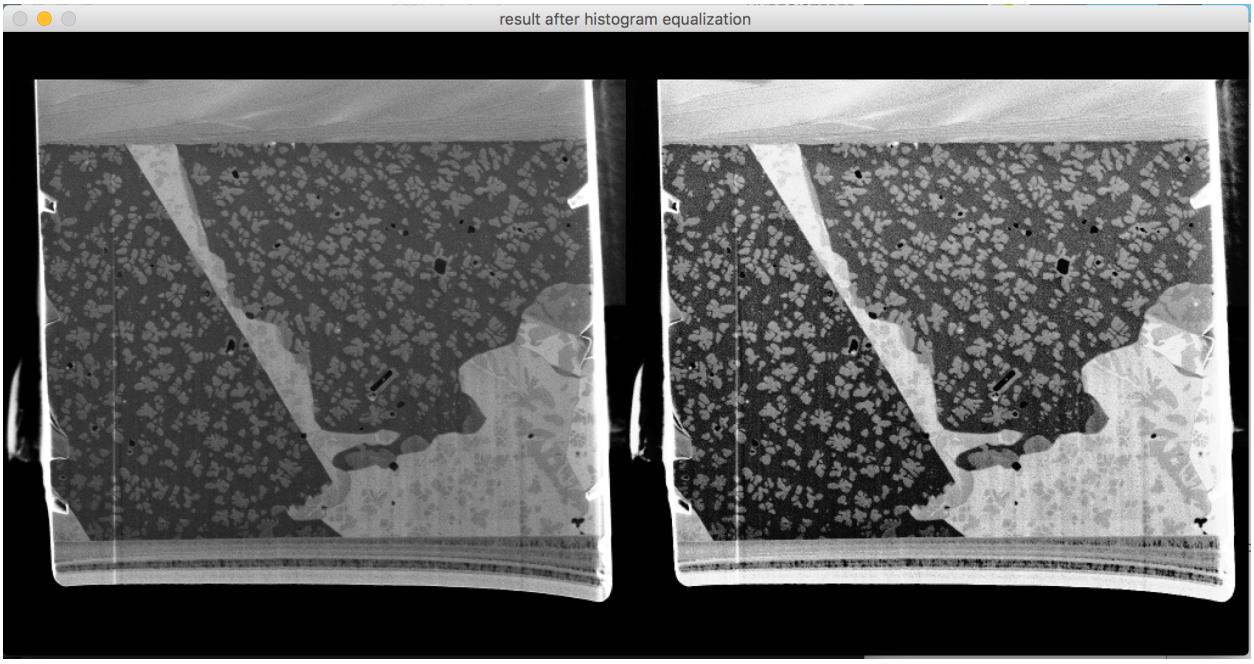
1) Calculate Histogram code
   - to get a sense how the distribution looks like;



2) OpenCV has better built-in functions;
   - 3 lines of code gives out pretty decent results
   2 example pictures attached below;

Next step:
   - tie this module into top_module; so all the aligned images call this function for histogram equalization;


result after histogram equalization


result after histogram equalization