To Do List for Coding:

1. Custom Particle Tracking
   1. Read Image Metadata
      1. anything and everything we can get out of the metadata
         1. title
         2. scale info
         3. date created (for future reference to identify original images)
   2. Identify Objects
      1. Need to clear noise from image
         1. CANNOT be based solely on intensity as SNR is really low on upper frames
      2. X, Y, Z positional information (subpixel)
         1. positioning should be based on intensity, can use filtered image to do initial localization
         2. need to find best algorithm to make subpixel estimates
         3. x and y localization first, and then z, all based on center of mass estimates of intensity
         4. Intensity information for Z localization must reference the original image stack, not a filtered one (see section 2, ‘Track Z-Displacements’)
      3. XY Plane Outlines
         1. Outlines may eventually be used to recreate 3D volumes as well as to identify strain in the markers
         2. For now, these will server primarily as reference information to assess the quality of the tracking.
         3. Eventually would like to be able to assign and reassign indices to outlines to associate objects between frames
      4. Intensity Values
         1. Don’t need to be stored
         2. Would like some output information about statistics
            1. Average feature brightest values based on distance from surface
            2. histogram of feature intensities
      5. Index Number ordered by row->column
         1. Or whatever ends up being most logical approach
      6. Assign a Quality
         1. How big is it
         2. How bright is it
         3. Does it sit in the array
         4. etc.
         5. Generate an initial score.
   3. Link objects from frame to frame
      1. Record ‘Trajectory’ Information
         1. Record displacement vectors between frames
         2. I would like to be able to determine why many objects appear to follow a curved trajectory using the current tracking system, I am not sure if it is real.
      2. Use trajectory information as second ‘Quality’ parameter
         1. Need a filter or reassignment of objects that appear in a low number of frames (some may be stray signals, some may be objects with large travel)
      3. Use Quality information to remove false positives
2. Track Z-Displacements
   1. Use intensity information to approximate reference state
      1. Use fits to approximate sub-pixel z location of feature centroids
         1. Will need trajectory information to identify intensity and shape to fit a center to the three-dimensional object based on intensity
      2. Use average fit around pillar of interest to create reference for deformed pillars
         1. The code needs to be able to identify what the original ‘unstressed’ position of the pillar would have been
            1. Find undeformed neighbors
            2. use the neighbors to build a plane that contains centers
            3. use the plane the approximate the unstressed center for deformed pillar of interest
3. Generate Strain Information from shear and normal displacements
   1. Update outputs to reflect both types of deformation
   2. Update outputs to reflect strain instead of displacement
4. Generate hexahedral elements using centroid information for reference and deformed cases
   1. Create input mesh file for FEBio
5. GUI
   1. Qualitative assessments of pre-processed/processed images
   2. Batch Processing
   3. Publishable and user-friendly code.
6. Update/Clean/Improve existing code