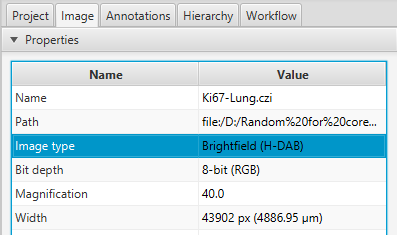
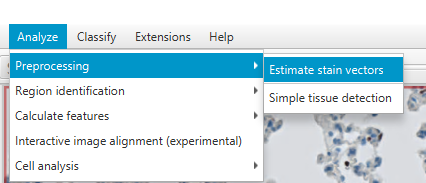
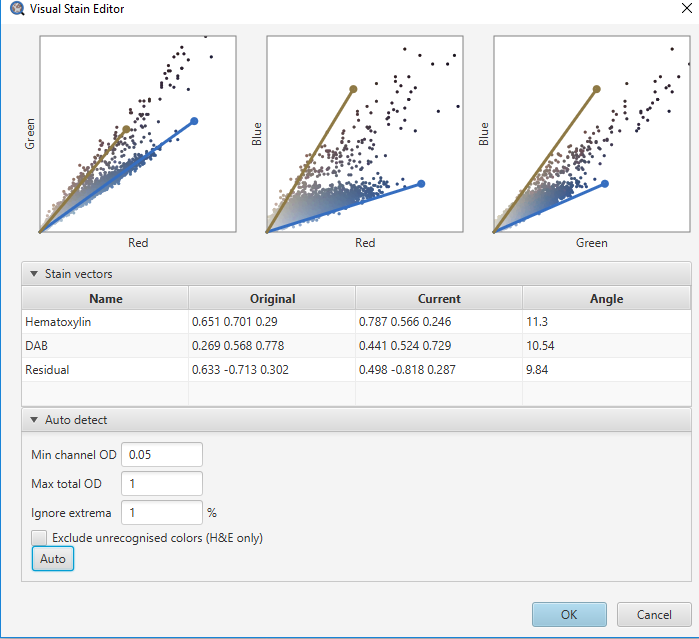
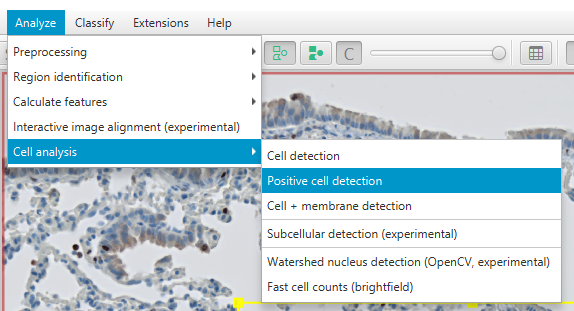
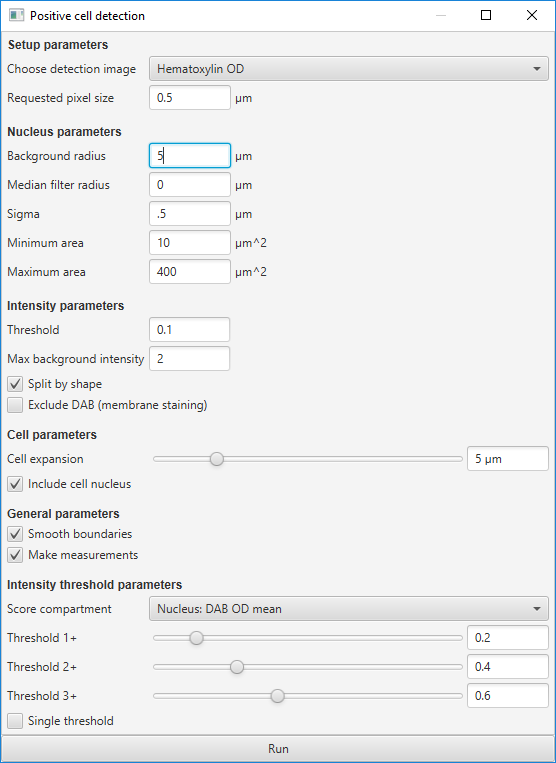
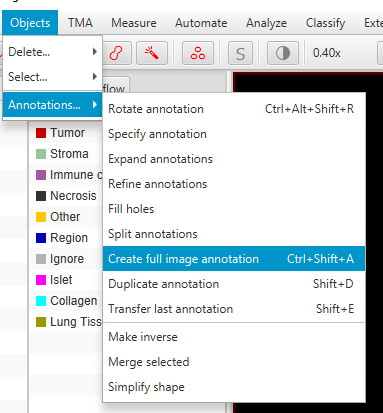
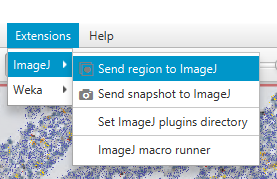
**Positive Cell Detection and H-Score**

Step-by-Step Protocol

**Goal**: Detect DAB+ cells in a tissue in a fast, robust manner

Assumes you are already familiar with projects, annotations, etc

1. Create a new project in any version of QuPath
2. Download Ki67-Lung.czi and add it to the project
   1. This is a cropped image of a mouse lung that was immunoperoxidase stained for Ki67, a marker of cell replication that should be localized to the nucleus. Nuclei are stained with hematoxylin (blue) and Ki67 is marked with DAB (brown).
3. Set the Image Type. Go to the Image tab > Image Type > Brightfield (H-DAB)  
     
   
4. Determine the stain deconvolution. Draw a rectangle ROI  around a small region that has some DAB+ cells, DAB- cells, and background. Then go to Analyze> Preprocessing > Estimate stain vectors.   
     
   It will ask you if you want to use the modal values as background- say Yes. Then, in the new window, click Auto and wait for the arrows to adjust to the data. Click OK, then give this set of stain vectors a name (default is fine).   
     
   
5. Back in the Annotations tab, select the drawn rectangle. Then go to Analyze > Cell analysis > Positive cell detection  
     
   This will bring up a window that allows you to determine the parameters for finding the nuclei, expanding the nuclei outlines to determine the full cells, and then labeling the cells as positive or negative for DAB.   
     
     
     
   The default parameter values are usually a decent starting place to detect cells. Press “Run” to have it detect and label cells in the rectangle region. If this takes too long to process, make your rectangle smaller. Then, adjust the parameters to improve the detection accuracy
   1. *Choose detection image:* Nuclei can be detected on the hematoxylin stain (typical) or the sum of all channels (unusual, but useful in some cases)
   2. *Requested pixel size:* the precision of the outlines. Larger numbers → faster, less accurate processing
   3. *Background radius:* size of the filter it uses for subtracting background from the chosen detection image. Should be a little bigger than your biggest nucleus
   4. *Median filter radius:* applies a median smoothing filter with the chosen radius before nucleus detection. 0 = no filter applied.
   5. *Sigma:* applies a Gaussian smoothing filter with the chosen radius before nucleus detection. 0 = no filter applied.
   6. *Minimum and Maximum area:* the smallest and largest objects that can be nuclei. Objects outside this range are discarded.
   7. *Threshold:* The nucleus detection threshold after all background removal and smoothing filters are applied. Lower → more things detected
   8. *Maximum background intensity:* Any regions with a calculated background higher than this threshold are likely noise/artifacts/problematic and will be discarded entirely.
   9. *Split by shape:* Check in almost all cases. Separates nuclei that are relatively round.
   10. *Exclude DAB (membrane staining):* Check if you know that your DAB staining is necessarily on the cell membrane and therefore anywhere that is DAB+ cannot be a nucleus. In this example, Ki67 is found in the nucleus, so do not check this.
   11. *Cell expansion:* All of the nuclei outlines will be expanded outward this distance to define the cell boundaries. They will stop when they run into another cell.
   12. *Include cell nucleus:*  Keep this checked. It will delete the nuclei if you uncheck it.
   13. *Smooth boundaries:* Keep this checked most of the time. It improves results while minimally affecting processing time.
   14. *Make measurements:* Useful if you are going to do any further processing on the cells after detection. Slightly increases processing time and file size, but it’s almost always worth doing.
   15. *Score compartment:* Tell it where in the cell you are expecting the DAB signal (only nucleus, only cytoplasm, or both), and if it should look at average or max intensity.
   16. *Thresholds:* Set the thresholds to determine 1+, 2+, 3+ cells (i.e, divide cells into negative, dim, medium, and bright). These are essential to calculating H-score.
   17. *Single threshold:* With this checked, it will only use the 1+ threshold, and will only tell you positive vs negative. No H-score will be calculated. Keep it unchecked for this exercise.
6. After you are satisfied with the positive cell detection parameters, you can delete the rectangle and the descendent cells. Draw a new box elsewhere in the image and double-check that the optimized parameters are still sufficiently accurate.
7. Annotate your entire tissue. If you are working with a whole slide, use Analyze > Preprocessing > Simple tissue detection. If you are working with a cropped portion of a slide (as in this example), use Objects > Annotations > Create full image annotation  
   
8. Go to the workflow tab, find the last (most recent) time in the command history you ran “Positive cell detection”. Double clicking that will open up the Positive cell detection box, with all of the parameters you set the last time you ran it.
9. Hit “Run” and it will detect cells in the entire image with the last settings. If you do not have the annotation selected, it will confirm that you want to process the annotation. This may take a few minutes.
10. In the Annotation tab, click on the full image annotation to see: 1) total number of cells, 2) number of cells at each threshold level, 3) Total H-score.
    1. H-score is a way of calculating how “positive” an area is, where 3+ are weighted more than 2+, etc. See [here](http://www.ascopost.com/issues/april-10-2015/calculating-h-score/). It is historically calculated manually (a pathologist will label every cell in a small area by hand), but doing it automatically enables you to use a much larger area, much faster.
11. Optional next step: Find a representative region. When you publish your amazing pathology results, you will likely want to make a figure that has a zoom-in on a small region that is “representative” example of the total tissue H-score. But how to find a region that is truly representative? You surely wouldn’t arbitrarily pick a pretty region that looks nice, right?
12. Download the file RepresentativeH\_allVer.groovy and drag it into the QuPath window. In the script window, hit Run>Run to make the script run.
    1. The output after a few minutes will be ~9 highlighted regions- 3 with particularly low H-score, 3 with average H-score, 3 with particularly high H-score.
13. For any region you would like to save and use as “representative example”, select the rectangle ROI, then go to Extensions > ImageJ > Send region to ImageJ  
    

Choose a downsample value of 1. From there you can save the image in your favorite format.

1. Advanced Optional Next Step: Practice Groovy scripting. Go through the “Advanced Scripting Exercises” document, which takes you through different ways of modifying the RepresentativeH\_allVer.groovy script. Lots of help can be found in the [QuPath Google Group](https://groups.google.com/forum/#!forum/qupath-users) (out-of-date), the [Image.sc forums](https://forum.image.sc/tags/qupath), [Svidro’s gist](https://gist.github.com/Svidro), or [Pete’s blog](https://petebankhead.github.io/).