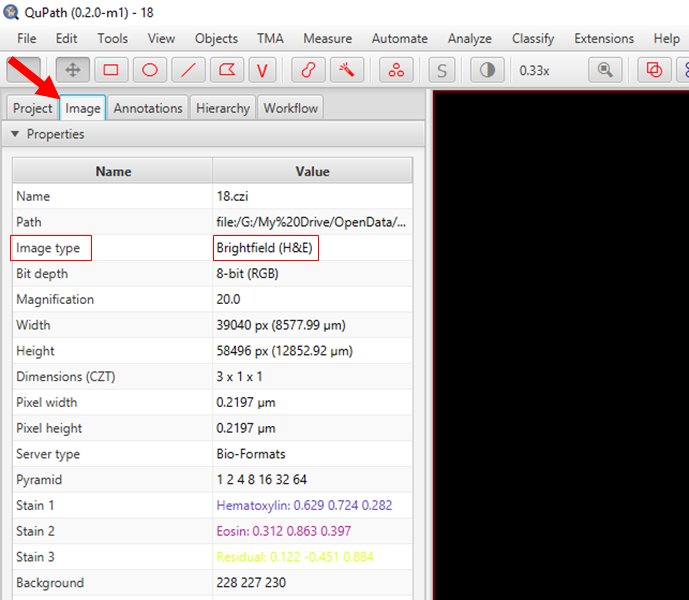
Exercise Overview

In this exercise you will learn how to classify cell detections to measure the tumor burden in metastatic lung melanoma model. We will walk you through the major steps that will be common for many analyses involving cell detection and classification on an H&E sample.

Exercise Goals:

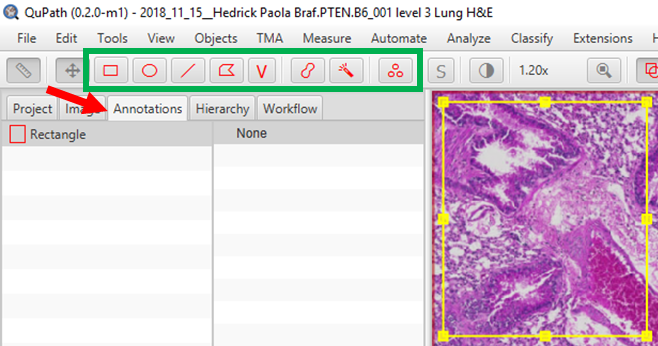
1. Set the stain
2. Stain deconvolution
3. Simple tissue detection
4. Cell detection
5. Annotate regions of similar cells
6. Classify annotations
7. Build classifier
8. Generate scripts to run throughout entire project



Set the Stain

Create a new project and add the images (5.czi and 18.czi) to it.

Click on the Image Tab on the left side and look for Image Type

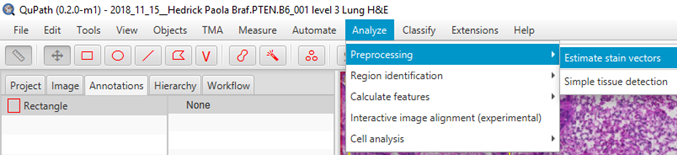
Double click on the Value and make sure that the image type is correct (in this exercise they are all H&E)

From there, click on the Annotations Tab located next to the Image Tab

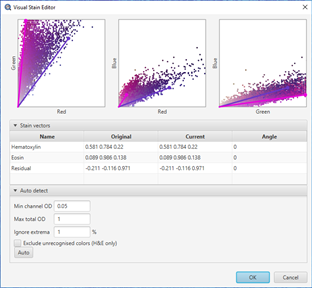
Select a shape from the panel of red tools

Draw a region of interest that encompasses all the different colors of the slide, but is not too large. Include some empty area too (this is your background).

Color Deconvolution

Next, click on the Analyze tab and select Preprocessing > Estimate Stain Vectors

Click Yes to accept modal values

Adjust RGB values for Hematoxylin and Eosin by either manually moving the stain vectors or pressing the Auto button, then hit OK.

QuPath will ask you to save the name of this stain vector, it’s best to assign it a new name (i.e. H&E tumor)

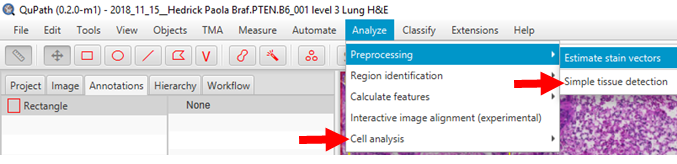
Now check the deconvolved (and some extra) channels by pressing numbers 1-9.

Simple Tissue Detection

Return to the Analyze tab > Preprocessing, this time selecting Simple Tissue Detection.

A window will open to a set of options for detecting the tissue within your image.

You can try running the default settings, and if you find that the lung tissue is not being properly selected, you can try increasing the threshold to 200 or 220, changing the minimum and maximum area of tissue being detected, or the requested pixel size.

Once the tissue has been detected, you will see under the annotations tab that the rectangle you drew is gone, and in its place is an annotation of the lung tissue.

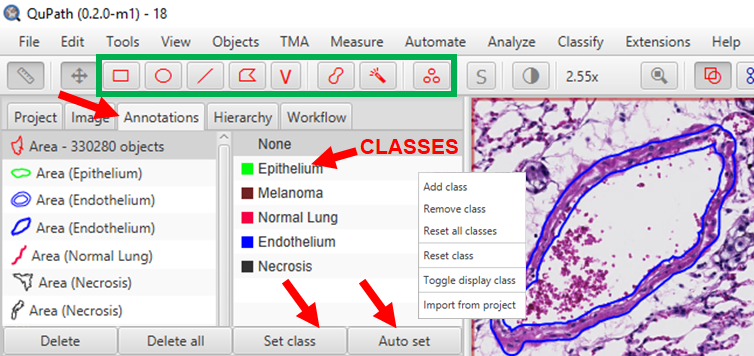
Cell Detection

Now that the tissue has been detected, we will automatically detect cells (objects) within the annotation. Proceed to the Analyze tab again, and select the Cell Analysis menu.

This will bring up a submenu of various methods to detect cells. Select Cell Detection at the top of the list.

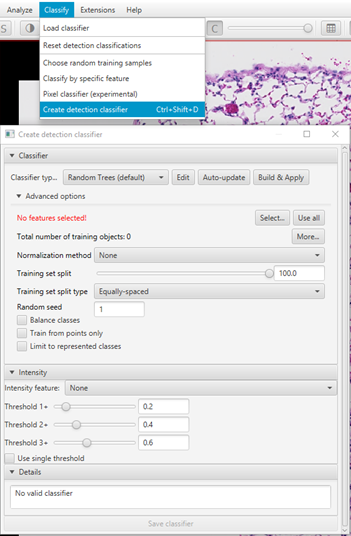
This window will have several parameters for detecting nuclei of cells. You can run the default parameters. Because these tissues have compact clumps of cancer, it might be useful to adjust the Sigma and the background thresholds in order to help break up detections of multiple nuclei and to detect more nuclei (or fewer false positives), respectively. Additionally, you can change the size of expansion for the cytoplasm around the nucleus can be changed to accommodate smaller/more tightly clustered cells.

Survey several areas of the tissue to make sure you are happy with the detections. If not, change the parameters and rerun. Have a look at the bottom of this document for a little discussion on cell detection. If you are working on an older computer, you might want to run the cell detection and classification in a smaller region of interest. This will run significantly faster.

Building a classifier

Now that we have cells detected within the tissue, we can start to classify them. Begin by identifying how many different regions of the tissue there are and creating classes for them. For this exercise we will have Normal Lung, Epithelium, Endothelium, Melanoma and Necrosis.

In order to set the classes, right click within the right-side of the annotations window. Choose Add Class, then type in the names of the classes above.

“ Once the classes are labeled, create an annotation for each of those classes. You can do this in 2 different ways. You can create an annotation first by selecting a drawing tool in the toolbar, then drawing on the tissue to select the cells that are part of the same type of tissue. Once the cells are selected, you can select the class they belong to and click on the Set Class button in the annotation window. You should then see the number of cells (or objects) that are inside that annotation.

Repeat this process for each class. Try to annotate the same number of cells for each class, and start with a small number of cells, around 100.

A second method for adding annotations is to select the class first, then press the Auto Set button in the annotations window. This will automatically add cells to the class, but be careful when you are moving between classes so that you don’t add objects to the wrong class as this will confuse the classifier.

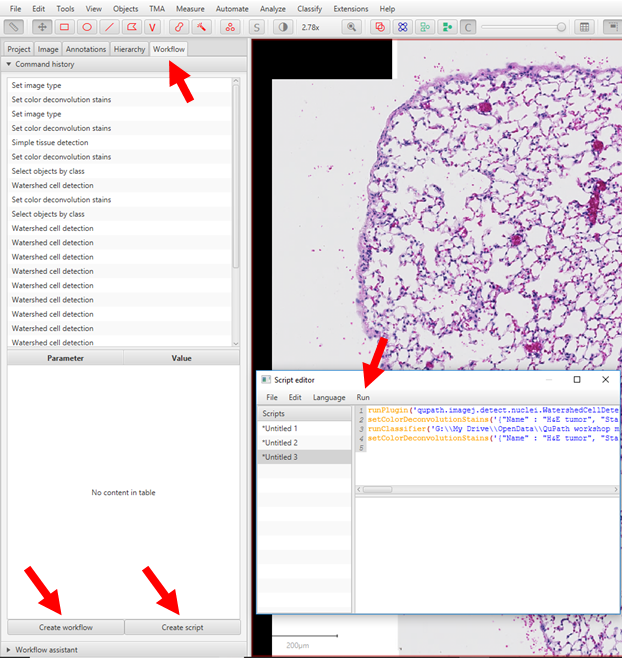
Once all the classes have a similar number of annotations, click on the Classify button next to the Analysis tab and select Create classifier from the submenu.

In this new window, you will have many options to try. To begin with, try running the classifier with the default settings by clicking on the Build & Apply button. If the classifier is not classifying the cells within the tissue properly, you can try adding objects to each class with additional annotations, using different classifiers, or different features under the advanced section, or.

If additional annotations are added, you can click on the Build & Apply button to rerun the classifier. If you would like the classifier to automatically update itself, you can click on the Auto-update button before adding new annotations. You will see the classifications of the tissue change after adding each new annotation.

Generating scripts to run throughout the project

There are a couple ways to generate scripts from the image we just analyzed. Clicking on the Workflow tab will bring up the command history that was used.

At the bottom there are 2 additional buttons, Create Workflow and Create Script. In either case, a script will be generated that can be edited if you want to remove commands or add commands.

It can be saved as a .groovy file (the language that QuPath uses).

After creating the script, there is a Run tab with a drop-down menu of options for how to run the script. “Run For Project” can be selected to run the entire workflow script automatically through the entire project.

Afterwards, it is a good idea to check as many files as possible to verify that the script ran as you expected and that you are obtaining similar accuracy of cell and tissue detection and classification between images.

**Want to see how your analysis compares with other people? Add your numbers to this sheet.**

[**https://docs.google.com/spreadsheets/d/1pzs\_IsqcRLDiVZnRAZbKh2iQbbpkRk1b4x0lE1cw1So/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/1pzs_IsqcRLDiVZnRAZbKh2iQbbpkRk1b4x0lE1cw1So/edit?usp=sharing)

**Notes**:

*Stain deconvolution* – as you may have noticed, using different starting regions will give you different stain vectors. Try to optimize this step if you don’t get a good separation between H & E. Check Pete’s YouTube tutorial on how they work in detail.

*Cell detection*– try to spot what kinds of cells are detected properly and where cell detection makes mistakes. You will almost always find some errors, but with enough practice you can tune it to be pretty accurate. It is very helpful to critically evaluate your analysis to make sure that your final conclusion is consistent with the data. If the outlines of your cells of interest are very fragmented, you may end up over or underestimating their number. On the other hand, if you simply want to measure the total area of given class, the results of adding up smaller fragments might still be accurate. See Pete’s tutorials for more details.