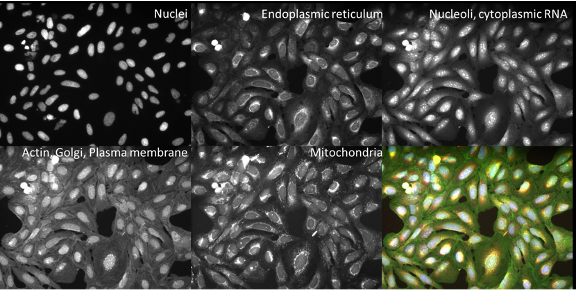
# CellProfiler Workshop Practice 2: Segmentation and organelle analysis

**Background information:**

The images in this experiment come from the [Broad Bioimage Benchmark Collection.](https://data.broadinstitute.org/bbbc/BBBC022/) They are fields of U2OS cells imaged in five channels (Cell Painting assay; see Gustafsdottir et al., 2013)



**Goals of this exercise:**

This exercise will give you practice finding segmentation parameters for larger “parent” objects (nucleus, cell, and cytoplasm) and show you ways to pull out smaller features in your image by segmenting organelles within the cells and nuclei. You will also be shown how to use RelateObjects so that you can relate the average counts, distances, and measurements of the smaller “child” organelles to their larger “parent” objects (i.e., cell and nucleus).

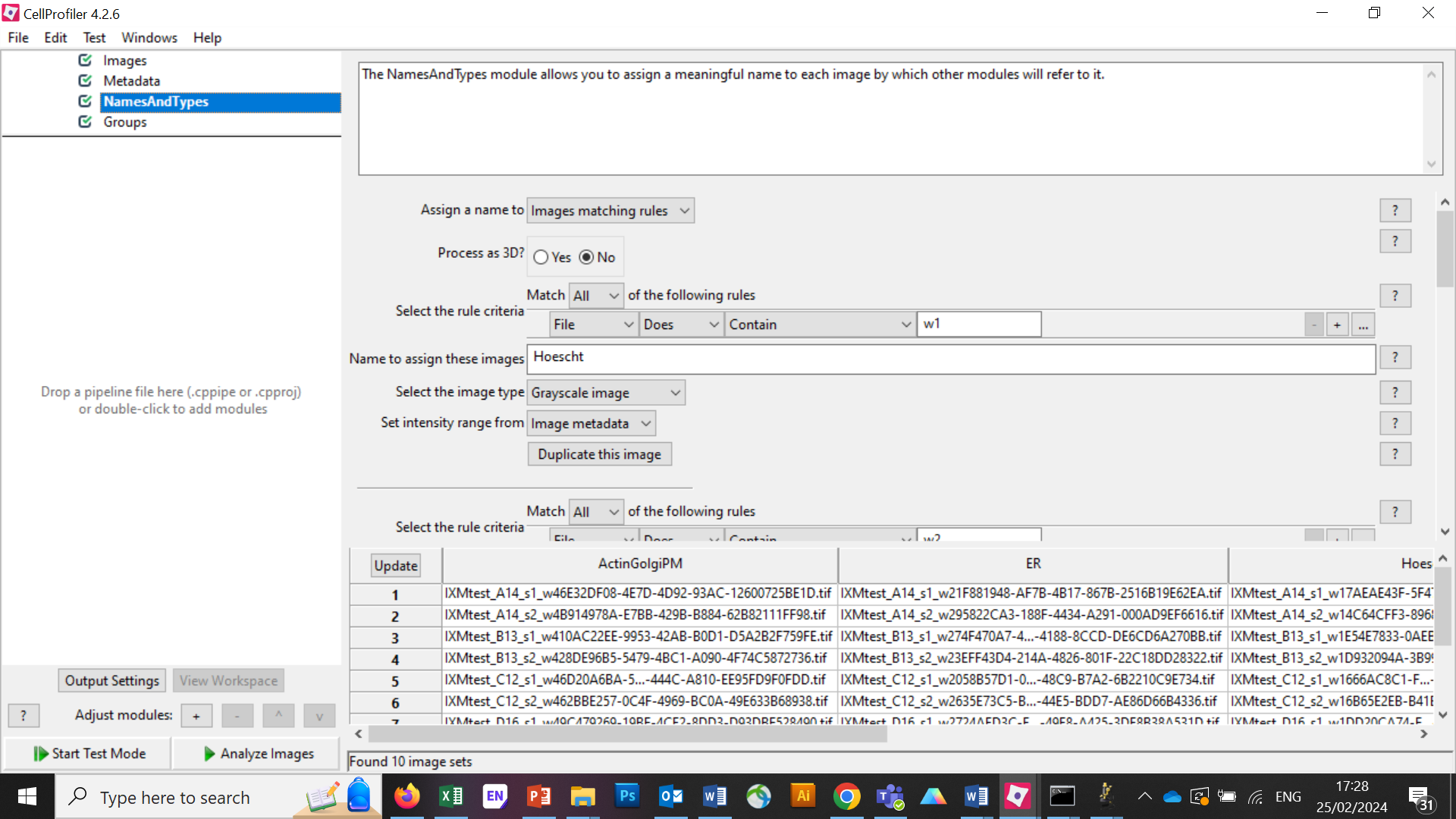
**Materials necessary for this exercise:**

The images are contained in the **CPW2\_Example\_5channel** folder; these 50 images (10 sites imaged in 5 channels) represent 5 mock treated wells from a single 384 well plate experiment.

**Exercise instructions:**

# Set up the input modules

* Drag and drop the **CPW2\_Example\_5channel** folder into the Images module.
* Under NamesandTypes, assign names with images matching rules for the 5 channels. In this example, Hoescht = w1, ER = w2, Nucleoli = w3, ActinGolgiPM = w4, Mitochondria = w5.
* Use update to test the images are identified correctly.

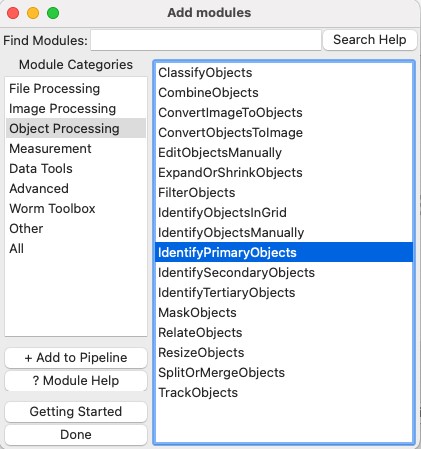


# IdentifyPrimaryObjects – Nuclei

Next, we’ll take a first pass at identifying nuclei and cells in our initial image.

Add an IdentifyPrimaryObjects module (from the ‘Object Processing’ module category). Do this by clicking on the ‘+’ sign in the bottom left corner of the CellProfiler window, which will pop up a small window called ‘Add modules.’.

Tip: You can also use the search bar at the top of the ‘Add modules’ window to search all modules by name.



Create objects called Nuclei by segmenting on the Hoechst channel.

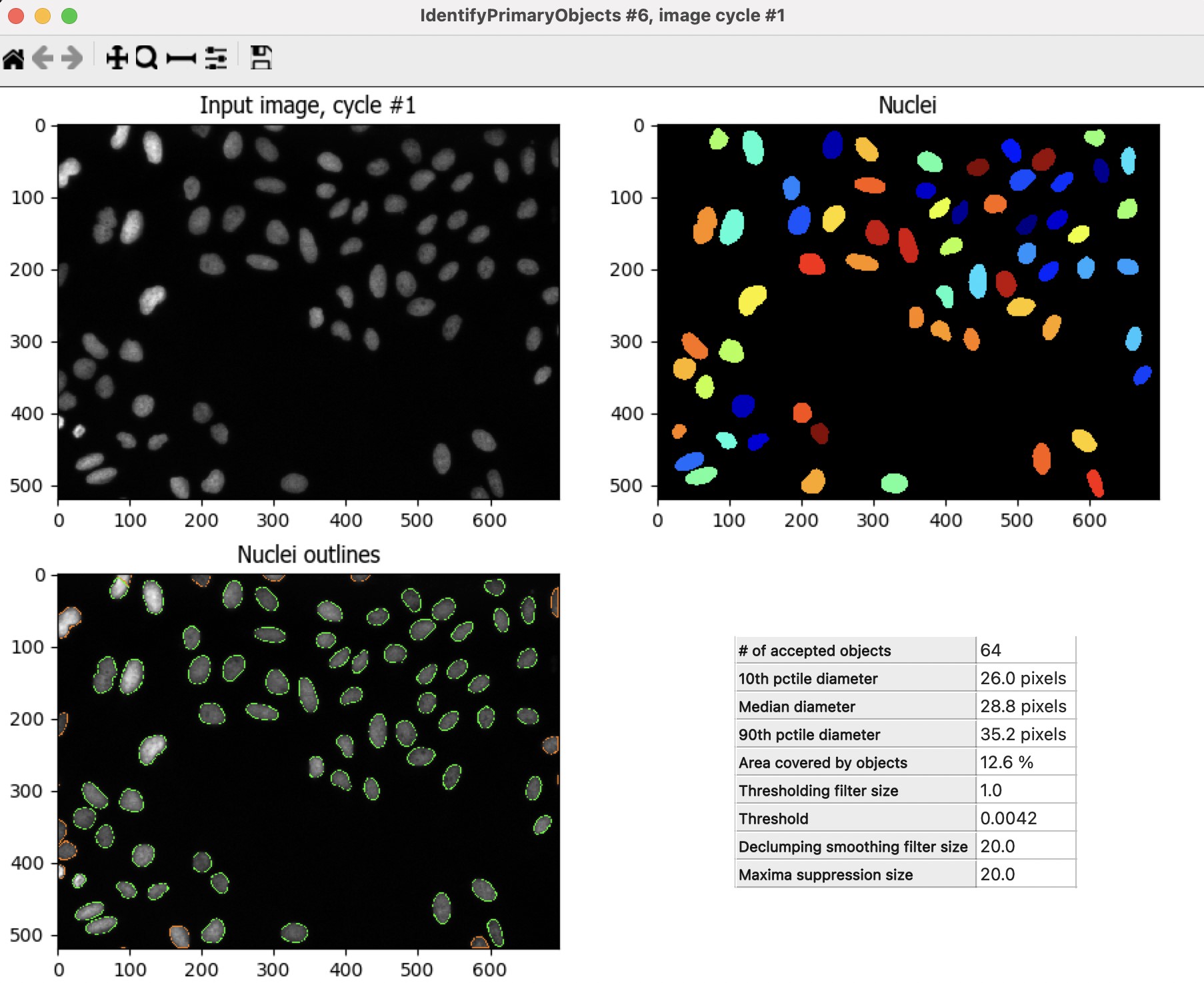
Select ‘Hoechst’ image as your input image from the drop-down menu.

Change the name of the output objects to ‘Nuclei’.

In Test Mode, hit’ Step’ to run the module. How does your segmentation look?

On the outlines display pane (bottom left) you can see three different colors; green is for accepted objects, orange for objects touching the border, and pink for objects outside the diameter range.

On the table pane (bottom right) there is useful information that you can use to adjust your segmentation settings, like the median diameter, and the threshold.



Use the magnifying glass at the top of the window to zoom in on an area that was segmented poorly.

**TO DO**: Improve your segmentation of nuclei:

Select ‘Yes’ for the ‘Use advanced settings?’ option, then change some of the parameters: Adjust the threshold method, may lead to better (or worse!) results. Adjust the declumping settings.

Hit ‘Step’ to rerun and see how the changes affect the segmentation.

Adjust the segmentation parameters until you feel you’re ready to move on to identifying the cells around the nuclei; the identification should be good but doesn’t need to be perfect before you move on.

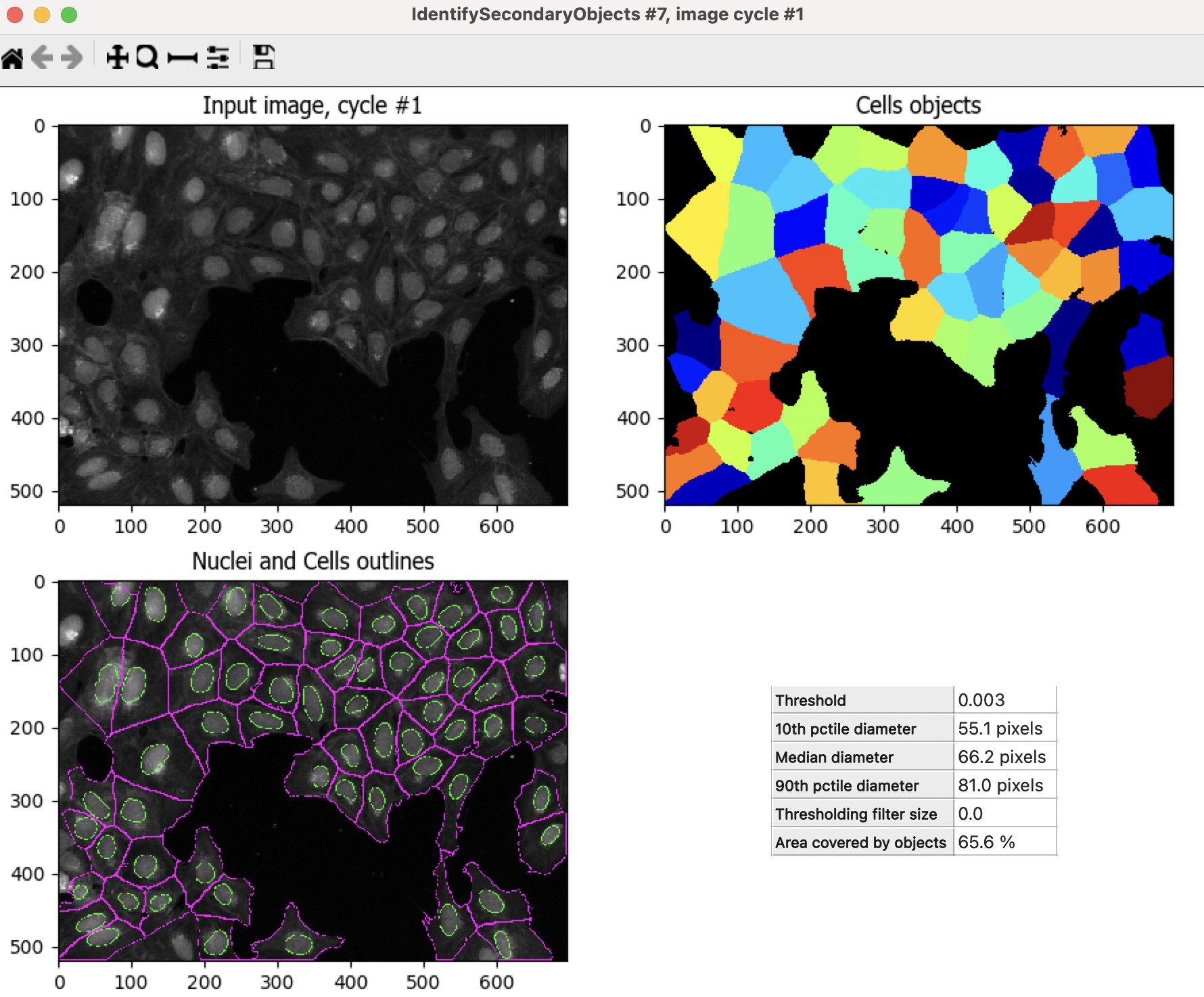
# IdentifySecondaryObjects – Cells

After the IdentifyPrimaryObjects, add an IdentifySecondaryObjects module.

Create an object called Cells that is seeded on the Nuclei primary objects that you just created; select the ActinGolgiPM image for your input image, nuclei for input objects and change the name to ‘Cells’.

Hit’ Step’ to run the module. How does your segmentation look?

For this module the outline colors correspond to the object seed (green-nuclei) and the segmented objects (pink-cell)



**TO DO**: Improve cell segmentation

Examine the segmentation and adjust the segmentation parameters until you feel you’re ready to test them on another image; they don’t need to be perfect.

Adjust the threshold method.

Test the effects of using the various methods for identifying secondary objects (Propagation, Watershed-Image, Distance-N, etc) and, if using Propagation, the regularization factor.

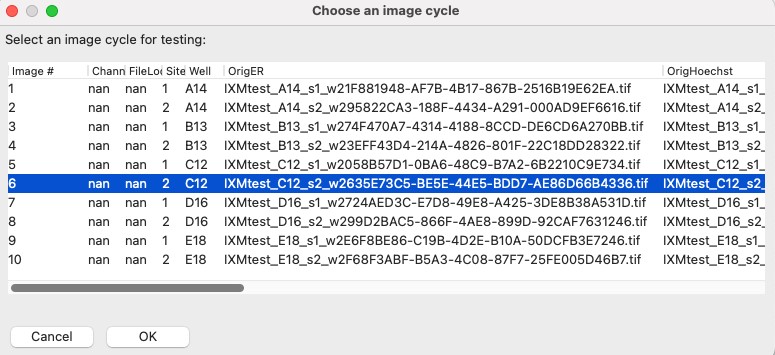
# Test the robustness of your segmentation parameters across images.

It’s (relatively!) easy to come up with a good set of segmentation parameters for a single image however we aim to create a set of parameters that can segment cells on all the images on an experiment.

To test the parameters, there are two options to change the image you are working on in Test Mode Click on the ‘Next Image Set’ at the bottom left corner, or

Go to ‘Test’ on the top menu bar → Choose Image Set to bring up a list of the images in your experiment, select the image you want to test, and press the ‘OK’ button.

Tip: you can also use the Test menu to choose a random image set



Then run that image in test mode for your first 3 modules (through your IdentifySecondaryObjects step).

You can do it by clicking the Step button, or

You can add a pause button on the module after IdentifySecondaryObjects and hit ‘Run’, this will run all modules before the pause.

Examine the output – did your nuclear and cellular segmentation hold up compared to the first images you looked at?

🔴 **TO DO**: Adjust the parameters to get comparable results to the first image. Once your segmentation is good, try it on another image.

# IdentifyTertiaryObjects - Cytoplasm

After the IdentifySecondaryObjects module, add an IdentifyTertiaryObjects module.

Create an object called Cytoplasm using the Cell and Nuclei objects you’ve created.

Select the larger and smaller identified objects from the drop-down menu.

Change the name of the objects to be identified.

‘Shrink smaller object prior to subtraction?’ should both set to ‘No’.

# Examine the steps used to segment the Nucleoli

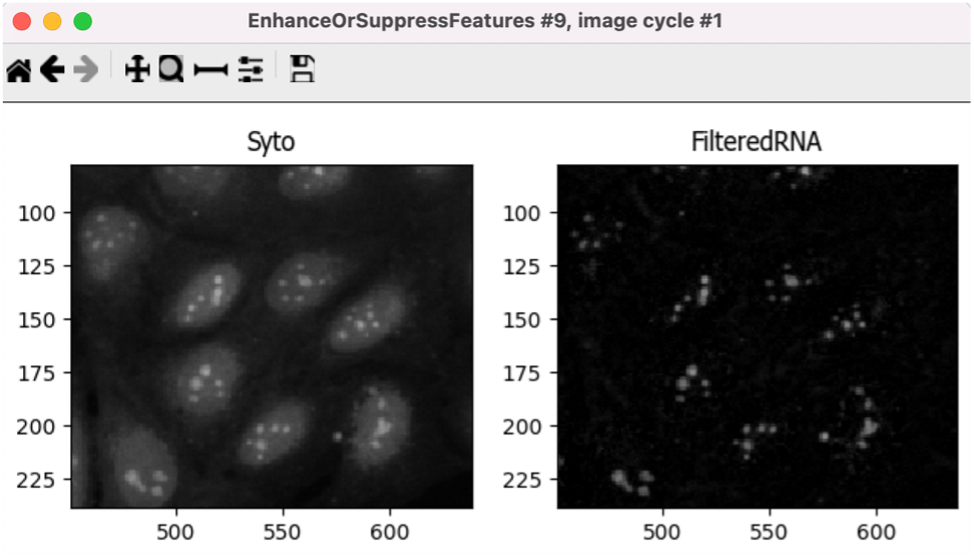
So far, we have used untransformed images for object detection, but not all objects can be segmented from raw images. CellProfiler contains a variety of image processing modules that can aid segmentation. For this exercise, we will use two such modules, but there are other ones you can explore.

The next 3 modules have to do with the creation of the Nucleoli objects. Look at the output from each to see how the image is transformed to aid in segmentation.

After the IdentifyTertiaryObjects module, add an EnhanceOrSuppressFeatures module.

EnhanceOrSuppressFeatures is a module that helps enhance parts of an image- in this case, punctate objects or ‘Speckles’. As we are looking for nucleoli, we apply this to the Nucleoli channel image and call the output ‘FilteredNucleoli’.

* **TO DO:** Enhance nucleoli spots.
* Change the input image from the drop-down menu to ‘Nucleoli’.
* Change the name of the output image to ‘FilteredNucleoli’.
* Change the feature size to see how this affects the output and find a value that works well.
* See below for an example of results to aim for:

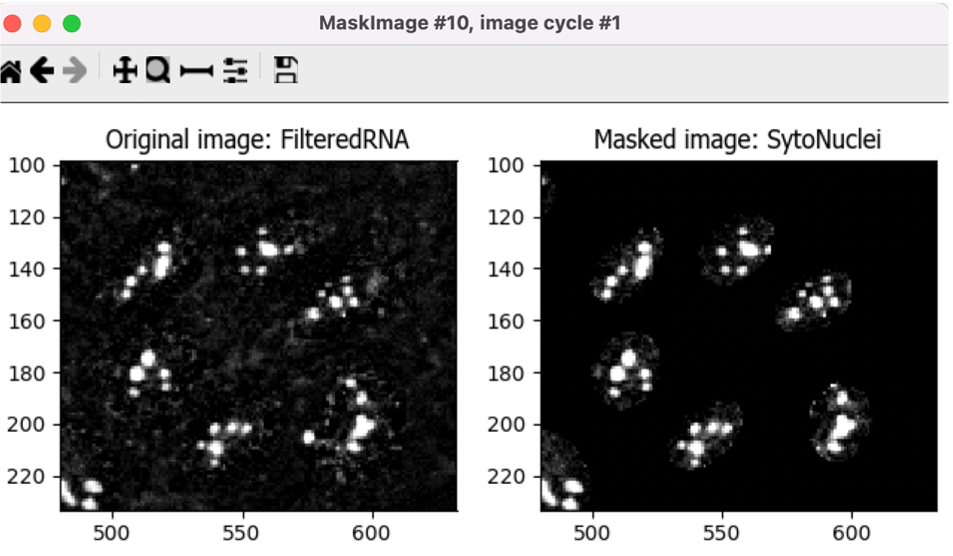


After the EnhanceOrSuppressFeatures module, add a MaskImage module.

MaskImage allows you to create a version of the ‘FilteredNucleoli’ image called ‘NucleoliinNuclei’ where all the pixels except the ones you specify are set to an intensity of 0. In this case, we set to 0 any pixel not inside a nucleus. By doing this, we can decrease the likelihood of detecting cytoplasmic RNA dots.

# 🔴 TO DO: Mask the Nucleoli image to show only the ‘Nuclei’.

* Change the input image from the drop-down menu to ‘FilteredNucleoli’. Change the name of the output image to ‘NucleoliinNuclei’. Use the objects ‘Nuclei’ as a mask.
* See below for an example of results to aim for:



IdentifyPrimaryObjects is used to find the Nucleoli. This is a Primary object segmentation because we are not using another object as a seed (i.e., starting point), and are only segmenting based off the intensity in our ‘NuceloliinNuclei’ image.

# 🔴 TO DO: Segment nucleoli

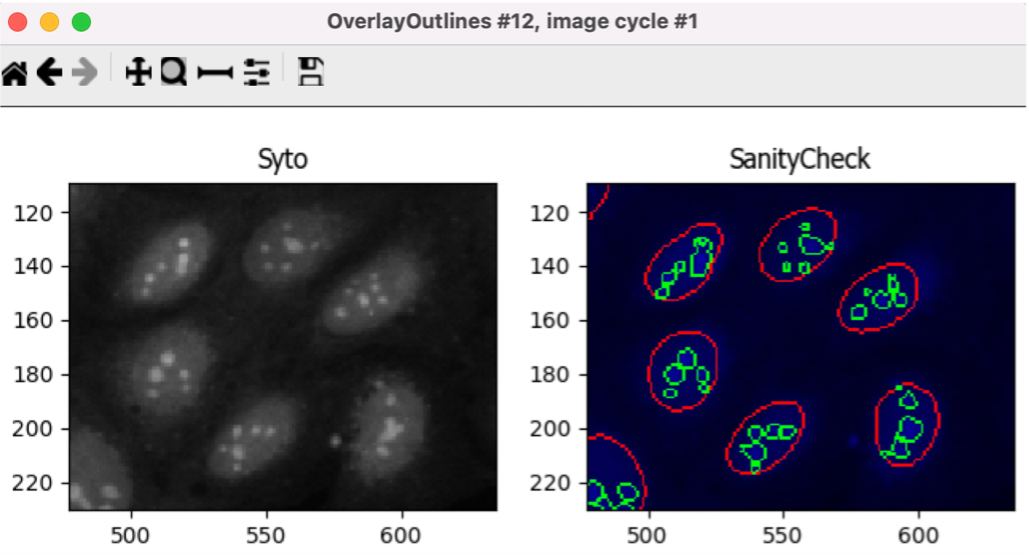
Change the input image from the drop-down menu to ‘NucleoliinNuclei’

Change the name of the objects to ‘SegmentedNucleoli’

Adjust the segmentation parameters until you are satisfied with the segmentation results.

🔴 **TO DO**: Add an ‘OverlayOutlines’ module at this point to overlay the identified nucleoli on the original ‘Nucleoli’ image to assure yourself that the segmentation not only matches the speckle enhanced ‘NucleoliinNuclei’ image, but also looks accurate on the unprocessed image as well. This is not strictly necessary but can be a nice “sanity check”.

* Goal: display outlines of your nucleoli and your nuclei on the unprocessed ‘Nucleoli’ image.
* Bonus points, can you add a step to the pipeline to save the “sanity check” image?
* Here’s an example of what that could look like (red=nuclei, green=nucleoli):



# TO DO: Add measurement modules to your pipeline.

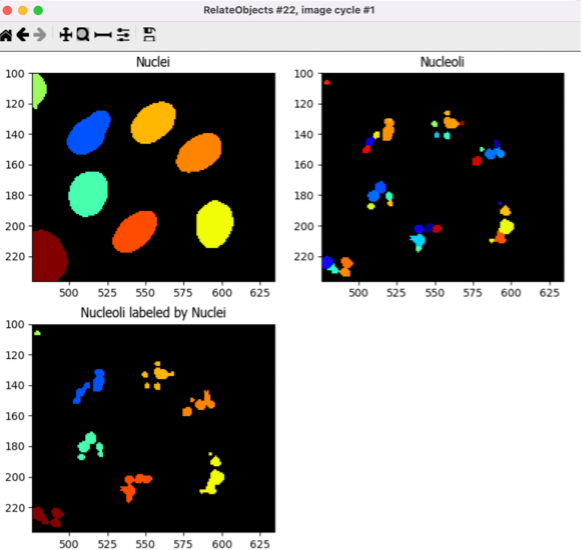
After your segmentation of the nucleoli, add as many object measurement modules as you would like, we have added a MeasureObjectIntensity.

Some suggested modules to add: MeasureObjectSizeShape, MeasureObjectIntensity, MeasureGranularity, MeasureObjectNeighbors.

Which objects do you think would be valuable to measure with each of these modules? Which channels would you measure your objects in? Note: While MeasureCorrelation, MeasureTexture, and MeasureObjectIntensityDistribution can produce valuable data for downstream profiling, they can be memory-intensive and/or slow so should not be added for this example pipeline in the interest of pipeline runtime.

# RelateObjects.

🔴 **TO DO:** Add a RelateObjects module and configure it to relate ‘SegmentedNucleoli’ to ‘Nuclei’.



Relating the objects allows you to create per-parent means (e.g., for this cell, what is the average size of an individual mitochondrion) and calculate distances from the child objects to the edge and/or the centre of the parent (e.g., how far is each nucleolus from the centre of the nucleus).

# Run the pipeline.

* Add an ExportToSpreadsheet module at the end and select which measurements to export.
* Exit test mode.
* Click on ‘Output Settings’ button at the bottom left corner.
* Change the default output folder to Elsewhere and send to a folder named ‘CPW3’ on Desktop.
* Click on ‘Analyze Images’ button at the bottom left corner.
* Explore the spreadsheets created for each object.