# Discovering drugs by using mixed cell cultures

Mark-Anthony Bray, David Logan and Anne Carpenter

## **Background**

#### The problem

To bring a drug to the point where it can be used in the clinic, assessing the ability of the compound or small molecule to actually hit the cellular target is only part of the development process. A key component of the process is assessing drug safety, that is, the impact of potential drug candidates on human cells. A drug that is determined to be toxic *in vivo* is clearly disqualified as a potential candidate. The problem is: how can such adverse effects be determined as early in the drug development process as possible?

Drug toxicity is especially important considering that it takes 10 to 15 years from the time a drug is discovered until it is administered to patients at the bedside. Add to this the fact that it takes \$800 million to \$1 billion to research and develop a drug, and for every 5,000-10,000 compounds that are considered for investigation, only one makes it through the FDA-approval process. The quicker such a compound can be ruled out due to toxicity, the less time and money is wasted in investigating a drug candidate, only to have it fail out at a late stage during clinical testing.

#### Traditional approaches

Typically, drugs are tested for toxicity by applying them to cells grown *in vitro*, that is, in an artificial culture system rather than situated in the human body itself (*in vivo*). There are a number of artificial cellular systems that are in use for drug discovery for various purposes, but one that is extremely common is the use of cells derived from the liver. The liver is a key organ in drug discovery since any compound in the bloodstream will eventually make its way to the liver for detoxification; toxic drugs are capable of poisoning the liver, causing harmful or possibly lethal side effects.

Rather than injecting humans or animals with the drug in question, creating a culture system of liver cells (or *hepatocytes*) to act as a surrogate for the liver organ would greatly benefit the discovery process. Using hepatocytes would reduce the cost of animal maintenance, reduce the amount of compound needed in the study, and more precisely investigate the mechanisms by which a drug has an effect, toxic or otherwise. Ideally, the hepatocytes are isolated from human tissue (such cells are called "primary" cells) in order to try to make the *in vitro* system reproduce the *in vivo* environment, making an "apples to apples" comparison possible. However, the use of primary hepatocyte culture suffers from some important limitations:

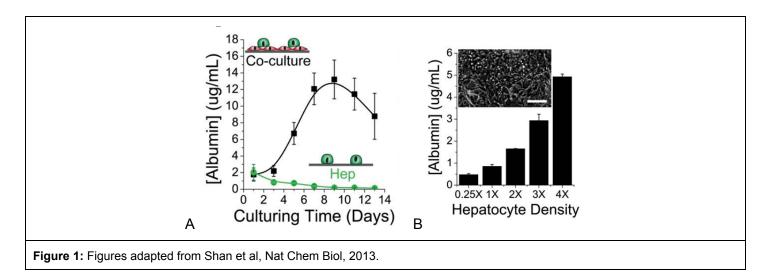
- Culturing these cells alone causes them to undergo a process known as de-differentiation, in which they change their structure and behavior and revert to a less-specialized form.
- Placing these cells in an environment without their normal non-hepatocyte cellular neighborhood changes their responses to chemical stimuli. Their behavior then becomes less likely to represent what actually occurs in vivo.

• The most important constraint is that primary hepatocyte culture lose their functionality drastically over time, in as little as 24 to 48 hours. This means that the time window in which studies can be performed is guite short.

For these reasons, finding compounds that enhance the growth and proliferation of human hepatocytes is of great importance in enabling drug research.

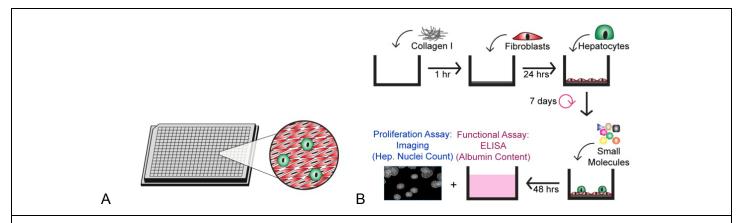
Using co-cultures to investigate renewable sourcing of hepatocytes

An approach that has shown promise in recent years is to co-culturing the hepatocytes with fibroblast cells. The rationale is that replicating the in vivo conditions by placing the hepatocytes in close proximity to the same cells they encounter in the human body will encourage the hepatocytes to retain their normal functionality. Indeed, research has demonstrated that hepatocytes grown this way will synthesize proteins and produce human metabolites in greater amounts and over longer culture times than hepatocytes with no supporting cells (Figure 1). We will use co-cultures as part of a large-scale, high-throughput testing (or "screening") assay of thousands of chemical compounds simultaneously.



## An image-based approach to hepatocyte proliferation research

To begin this experiment, we grow the cells in culture. We culture the cells in a small test tube (a "well"), in a multi-well plate that consists of 384 wells fused into a single plate (Figure 2A). Into each well of the plate, we first place a layer of extracellular matrix protein to act as a support for the fibroblasts. Once the fibroblasts are cultured on top of the protein and allowed to form a sheet, the primary human hepatocytes are cultured on top. We then dispense a different chemical compound in each well, and allow time for the chemical to induce the hepatocytes to proliferate. The cells are then then stained with Hoechst, a fluorescent dye that binds to nuclear DNA, and each well is imaged with an automated microscope (Figure 2B).



**Figure 2:** Schematic of automation of sample preparation. (A) Hepatocyte/fibroblast co-cultures are created by growing primary human hepatocytes (green) on a layer of fibroblasts (red) in 384-well plates. (B) To create the co-culture, the cells are first grown on a supporting protein layer, cultured for 7 days, and then treated with compounds for 48 hrs before images are acquired and analyzed for hepatocyte count, as well as an assay to check for hepatocyte function. Figures adapted from Shan et al, Nat Chem Biol, 2013.

## Activity #1: Calculate cellular measurements for classification

Your assignment is to perform a proliferation assay to determine which compounds produce more hepatocytes in the co-culture system. Thus far, attempts to screen for chemicals that enhance hepatocyte viability and proliferation have been limited due to the inability to produce hepatocytes in large quantities and the sharp decline in viability and liver-specific functionality when grown in culture. The co-culture system solves this problem, but counting the number of hepatocytes in an image by eye is not only tedious but simply not practical when the objective is to assess the effects of thousands of chemicals. Instead, you will have an automated microscope and a computer to perform this task.

In order to analyze the image data, you will use CellProfiler, an open-source image analysis package. This section contains instructions necessary to analyze images of hepatocyte co-cultures treated with different chemical compounds. Each well in the 384-well plate will contain a fluorescence image in which the DNA is stained such that it shines brightly (see Figure 3).

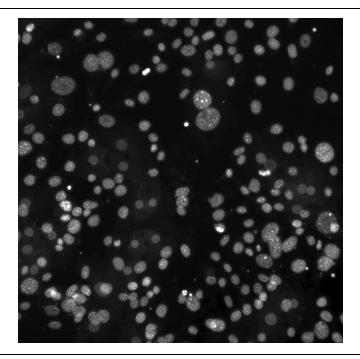


Figure 3: Example well with hepatocyte and fibroblast nuclei, both stained for DNA.

Once the nuclei are identified, the next step is to actually distinguish the hepatocytes from the fibroblasts. This step is challenging for a couple of reasons:

- 1. You will need to figure out the image features of each cell type that serve to distinguish one phenotype from the other.
- 2. Once you find the best image feature, you will need to decide on a cutoff value that will be used to distinguish the two phenotypes.
- 3. Even with working answers to items 1 and 2, all the nuclei will need to be scored as hepatocyte or fibroblast, and done so rapidly since there are many nuclei in many wells across the entire experiment.

This activity will focus on deciding which image features are most useful in distinguishing the hepatocytes from the fibroblasts.

Figure 3 below shows a number of close-up images of both hepatocytes and fibroblasts.

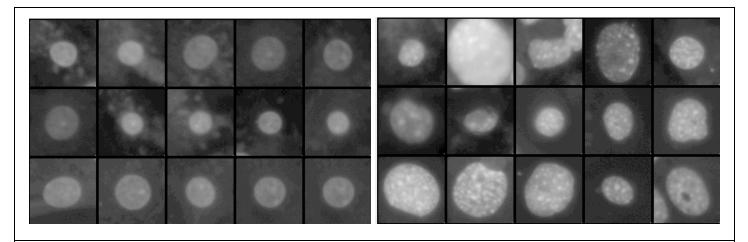


Figure 4: Images of hepatocyte (left) and fibroblast (right) nuclei. Both panels are shown with the same spatial scale.

In the world of high-content screening, there are a large number of cellular measurements that can be extracted on a per-cell basis, any or all of which can contribute in describing the phenotype of a cell. Some examples of image-based features include the following:

- 1. *Morphology:* This describes some aspect of an object's *shape*. Some commonly used quantities are:
  - a. Area: The number of pixels in one object (i.e., cell).
  - b. Perimeter: The total number of pixels around the boundary of one object.
  - c. Eccentricity: How round or oblong an ellipse would be if it fitted to that object. Equals 0 for a circle (the most round) and equals 1 for a line segment (the most oblong).
  - d. Major axis length: The length in pixels of the long axis of an ellipse that best fits an object.
  - e. Minor axis length: The length in pixels of the short axis of an ellipse that best fits an object.
  - f. Form factor: Describes how round or convex an object is. Computed as a ratio of the object's area to the squared perimeter. Equals 1 for a perfectly circular object and < 1 for an object with a less circular shape.
- 2. *Intensity:* These quantities are related to the amount of fluorescent marker at a given pixel location, or in an object overall. Some examples of intensity features include:
  - a. Integrated (total) intensity: The sum of the object's value at each pixel, which is proportional to amount of labeled substance.
  - b. Mean, median, standard deviation intensities
  - c. Lower/upper intensity quartiles
- 3. *Texture:* These quantities determine whether the staining pattern is smooth or coarse at a particular spatial scale. This can be calculated as a co-occurrence matrix representing how often pairs of pixel with specific values and at a given offset occur in an image, and then extracting statistical measures from this matrix. Examples of textural features include:
  - a. Variance: The dispersion around the mean between a pixel and its neighbor over the whole image.
  - b. Contrast: The intensity contrast between a pixel and its neighbor over the whole image. Evaluates to 0 for constant image.
  - c. Correlation: The linear dependency of a pixel to its neighbors over the whole image. Evaluates to 1 or -1 for a perfectly positively or negatively correlated image.

- 4. *Clustering/adjacency:* These quantities characterize the spatial relationships between objects. Some examples of clustering features include:
  - a. Number of neighboring objects.
  - b. Percent of each object's perimeter which touches a neighboring object.
  - c. Distance to the nearest neighbor.

**Question 1:** Looking at the example images (Fig. N), what qualitative features seem to distinguish one cell type from the other?

- Open CellProfiler from the Start Menu (Windows) or the Applications folder (Mac).
- The CellProfiler interface will appear, with a welcome screen to the right (Figure 5).
  - You can dismiss the welcome screen to continue, but is useful for getting oriented to the interface at a later time.

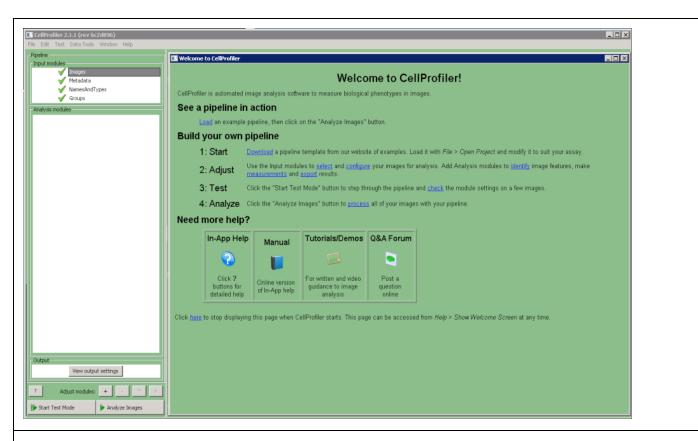


Figure 5: The CellProfiler interface upon startup (Windows screenshot)

- From File Explorer (Windows) or Finder (Mac), browse to the "pipelines" folder. Drag and drop the file 'activity1\_calculate cellular measurements.cppipe' into the "Analysis modules" panel on the left in order to load it.
  - NOTE: Upon loading, the modules in the pipeline will show red warnings since the inputs are not yet set. This will be taken care of in the next step.
- Click the "View output settings" button at the bottom-left of the interface.
  - o In the module settings panel on the right, click the folder button to the right of the "Default Input Folder" box. Browse to select the location where the file "filelist\_all\_nuclei.csv" is located. This file that provides the names and locations of the images that CellProfiler will use as input.
  - Now click the folder button to the right of "Default Output Folder" box, and browse to select your Desktop. This location is where your CellProfiler measurements will be saved.

The pipeline will use selected images from an actual screen in which the hepatocytes and the fibroblasts were co-cultured and then stained for DNA. In order to simplify the exercise, an expert manually drew the boundaries around the nuclei, and indicated which were hepatocytes and which were fibroblasts. We refer to this information as the "ground truth" for this set of images.

- Click the button labeled "Analyze images" at the bottom-right of the CellProfiler interface to start the pipeline. The software will identify all the nuclei in a well and make measurements on each.
  - (Windows only) A Windows Security Alert box may pop up asking for network access permission for CellProfiler.exe. Check the "Private networks" box, then click "Allow access".

The pipeline will run beginning with the first of 15 images, and the window will update 15 times, each time the next image is analyzed. The full run may take a few minutes.

### Description of the pipeline modules and their function

As the pipeline runs, you can't interact much with CellProfiler, but you can read a description of each module's function in the pipeline below and look at the corresponding module display window:

- LoadData: Loads each collection of images (DNA-stained fluorescence, plus images containing the boundaries for each cell) into CellProfiler.
- MeasureObjectIntensity:Measure a collection of intensity statistics evaluated per cell, for each image.
- MeasureObjectSizeShape: Measure a collection of morphological statistics evaluated per cell, for each image.
- MeasureTexture: Measure a collection of texture statistics evaluated per cell, for each image.
- OverlayOutlines: Superimpose the cell outlines on the original images to prepare them for output.
- Savelmages: Save the outline image to a file for later review.
- ExportToDatabase: Output the final set of image feature values to a special file for use with the CellProfiler Analyst software package.

Additional details are included in the "Module Notes" at the top of each module.

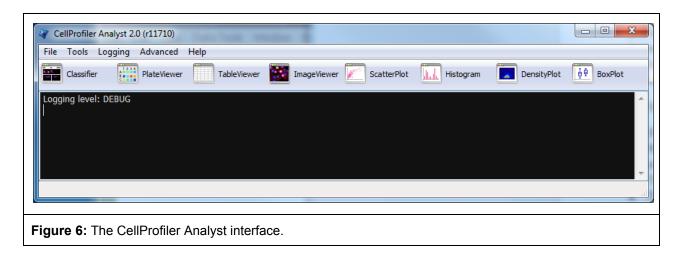
Once the pipeline is complete, you will have two new files in the Default Output Folder:

- A database file labeled "Activity2.db" (containing image feature values for each cell)
- A couple of folders containing images needed for previewing in the next step.
- Leave the CellProfiler interface open, as we will be coming back to it for Activity #3.

# Activity #2: Use supervised machine learning to classify the nuclei

In these experiments, the number of image features that are generated for each cell is quite large, so large, in fact, that it is impractical to assess by hand as to which ones are best-suited for distinguishing the phenotypes of interest. The next activity will involves training the computer to recognize hepatocytes from fibroblasts. Once you have trained the computer to do so (in a process called "supervised machine learning"), you will then apply that rules that the machine finds distinguishes the phenotypes to the data we used earlier, in order to count the number of hepatocytes versus fibroblasts.

- Open CellProfiler Analyst by double-clicking the icon from the Start Menu (Windows) or the Applications folder (Mac).
- When the program is started, it will ask to select a "properties" file. Select the file named
  "Activity1.properties", located on your Desktop (or wherever you set your Default Output Folder).
  The properties file was created by the ExportToDatabase module in your pipeline.
- The interface shown in Figure 6 will appear.



# Creating a collection of nuclei that will be used to determine the difference between "positive" and "negative" nuclei

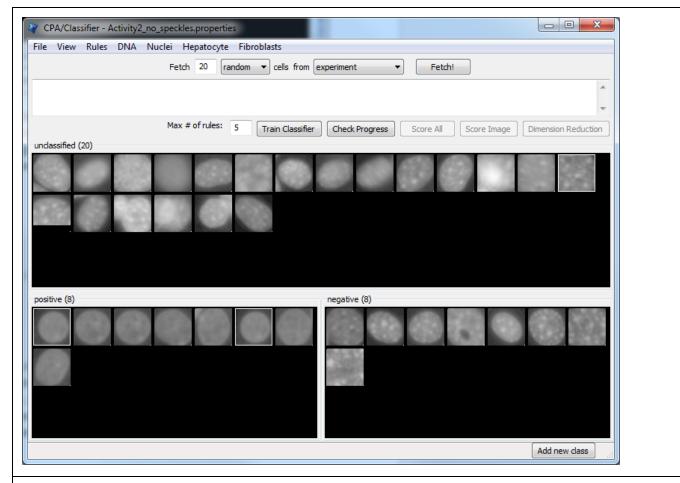
CellProfiler Analyst (or "CPA" for short) contains a machine-learning classification tool, which will allow you to distinguish different phenotypes automatically. In this case, we will "train" the classifier to recognize the hepatocytes ("positives") versus the fibroblasts ("negatives") by sorting a relatively small number of examples of each into bins.

• Select the Classifier icon in the CPA menu ( , 1st on left). The Classifier interface will appear, similar to that shown in the top of Fig. 7.

- Click on the "Fetch!" button, which instructs CPA to display pictures of the specified number of randomly selected nuclei from this experiment (in this case, the default of 20 nuclei). You will see a thumbnail image of these nuclei appear in the middle "unclassified" panel.
- From the "View" menu item, select "Image Controls" to enhance the contrast for easier viewing of the nuclei.
  - From the box that appears, select the "Log" radio button under "Contrast Stretch", then close the box.

Normally, the researcher would then simply drag-and-drop nuclei with the mouse from the "unclassified" panel into the "positive" or "negative" bins. However, since hepatocytes are fairly rare, it's easier to select them directly from the full image of the well rather than hope that they turn up randomly.

- Double-click any of the thumbnail images to bring up the full image that contains the clicked cell.
- Look for hepatocytes and fibroblasts within this image. You can refer to Fig. 3 above for examples of each type of cell.
- Once you find a fibroblast or hepatocyte, click on the cell center, and then "drag & drop" nuclei into their respective "positive" or "negative" bins at the bottom of the Classifier tool.
  - A cell that you consider to be a hepatocyte should be placed in the "positive" bin (You can rename the bins if you choose too, by right-clicking in the bin).
  - A cell that you consider to be a fibroblast should be placed in the "negative" bin.
  - If you're considering a cell and are not sure what type of cell it is, there is no need to classify it.
     However, all nuclei will be classified as one or the other at the end.
  - If you are unable to classify at least 10 clearly positive & 10 clearly negative nuclei, then double-click another cell in the "unclassified" bin to get the full image for that cell. Continue to look for nuclei, dragging & dropping them into bins, until you have enough nuclei in each bin.



**Figure 7:** The Classifier software showing 8 positive (i.e, hepatocyte) and 8 negative (i.e, fibroblast) nuclei. Twenty nuclei are unclassified and are ready for sorting.

- Once you have at least 10 nuclei in the positive bin and 10 nuclei in the negative bin, click the "Train Classifier" button.
  - Note that in this portion of the exercise, the cell images that are provided to you are a random sampling of the data. Thus, depending on which cell images are allocated to you, your sorting into these two bins will take varying amounts of time. Subsequently, the results of this portion of the exercise will not look the same from user to user.
  - We refer to this set of positive and negative nuclei you have assembled as the "training set."

# Reviewing the rules that CPA established (based on your training set) to classify positive and negative nuclei

The classification rules you will examine below are CPA's way of defining the image feature (and the cutoff values the image features need to have) in order to distinguish the positive from the negative phenotypes.

- Read the text that is now located in the text box in the upper half of the Classifier window. This
  text contains the rules CPA found based on the training set you provided to it.
- Each rule is in the form an "IF" statement evaluating whether an image feature is greater than some value.

• The main point to keep in mind is that the closer to the top of the list a image feature appears, the more significant it is in distinguishing the phenotypes.

#### Question 2:

- What is the top-most image feature that shows up in your classification rules?
- Is the top-most image feature one that you would expect to be the most significant one to use, in distinguishing the phenotypes (given your answer to Question 1 above)?

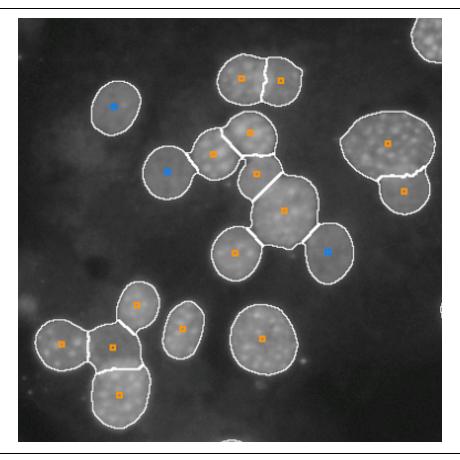
# Refining the training set by sorting more "unclassified" nuclei into the "positive" and "negative" bins

At this point, it is important to keep in mind that the CPA Classifier tool will pick whichever image feature is most significant in making its determination of positive versus negative (whether or not that image feature happens to be a physiologically relevant characteristic in the mind of the user).

For example, at this point (after only sorting 5 positive & 5 negative nuclei), you may notice image features called "Object\_Number" (the uninformative object number of each cell) or "Nuclei\_Location\_Center" (the cell position in the image) included in the classification rules. This indicates that the classifier is not well-trained, since these image features are not correlated with the phenotype we want to find. Whenever you find that the classifier is not well-trained, you need to either add more nuclei to the training set, or obtain more measurements from the nuclei.

You may also apply the rules to all the identified nuclei in an image, and use it to correct misclassifications.

- Double-click any of the cell thumbnails in the positive or negative bins.
- From the image that opens, click "Classify" from the menu, then "Classify Image".
- The nuclei will be color-coded according to their classification based on the current rules.
  - On Windows computers, to see what each color means, click the "Show controls >>" button at the bottom to reveal the colored class list.
  - On Macs, select "View" from the image menu, and then select "View cell classes as numbers."
     Then, to see what each number means, click the "Show controls >>" button at the bottom to reveal the numbered class list.
- Look for any nuclei that you feel are clearly misclassified.
  - For each of these nuclei that you find, click on it and drag-and-drop it into the appropriate bin.
- Click the "Train classifier" button. You will observe the rules change accordingly.
- Now, go back to the full image that you were using to correct misclassifications. Click "Classify", then "Classify Image" again.



**Figure 7:** A zoomed-in inset from an example image with the nuclei outlined in white, nuclei classified as positive (i.e, hepatocytes) marked in blue, and nuclei classed as negative (i.e, fibroblasts) marked in orange.

#### Question 3:

- After correcting observed misclassifications and re-training the classifier, what is the top-most measurement now? Did it change from what was before?
- If the answer to the above was "Yes", is the measurement one you would expect to be a significant one to use in distinguishing the phenotypes (given your answer to Question 1 above)?
- When you re-classified the image, did the classification appear to get better or get worse?

You now have your initial training set, and the rules that define the computer's first attempt at distinguishing the phenotype. Therefore you can now request that the computer fetch more examples of positive and negative cells, rather than randomly from the whole experiment. These new sample cells can be added to the corresponding bins, in order to improve the classifier's performance, with respect to distinguishing the two phenotypes.

- Change the number next to the word "Fetch" from "20" to "5".
- Click on the drop-down box labeled "random" in the fetch controls. Select "positive" from the drop-down list.
- Click the "Fetch" button to retrieve samples of what the computer thinks are positive cells based on the current set of rules. Refine your training set by doing the following:
  - o If positive cells are correctly fetched (true positives), drag and drop them into the positive bin.
  - If negative cells are incorrectly fetched (false positives), drag and drop them into the negative bin.

If you are not sure about which bin a cell belongs to, do not add it to the training set. Instead, click to select these ambiguous cells, and then press the "Delete" key to remove them from the analysis.

- Click the "Train classifier" button again.
- Repeat the above step until you have at least 20 cells in each bin.

Whatever approach you choose to obtain more positive and negative cells, the iterative procedure is the same: (i) Find rules; (ii) Obtain more cell samples of the desired phenotype; (iii) Correct misclassifications, or sort into appropriate bins; (iv) Go back to the first step and repeat, until the classifier displays the desired level of accuracy.

## Classifying all nuclei in the experiment

Once you are satisfied with the classifier's accuracy, it is ready to be applied to the complete image data set.

Press the "Score all" button. A dialog box will appear with scoring options; click "OK" to accept the
default settings and begin scoring. Every cell in every image will now be scored as positive or
negative by the classifier you built.

A "Hit table" window will appear containing the summarized scores for every image (Fig. 8). The total cell count is reported, as well as the number of positive and negative cells classified. The last column is the enrichment score, a weighted probability ratio expressing the likelihood of an elevated phenotype count.

- Click on the column header labeled "Enriched Score positive." (You can resize the hit table window, if this column is not visible). Clicking this header will sort the rows in ascending or descending order, according to the enrichment scores. Sort the column values so the order is descending, with the highest score at top.
- Double-click on the asterisk in the first row to the left of the first column ("ImageNumber") to display the corresponding full image for the top-scoring well.
- From the image that opens, click "Classify" from the menu, then "Classify Image" to see the final classification.

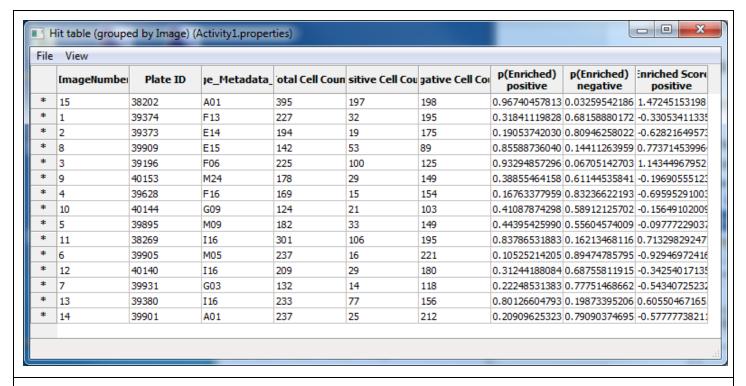


Figure 8: Hit table showing the cell counts and enrichment scores.

As mentioned earlier, the borders of the nuclei were identified manually by an expert. What's more, which nuclei are hepatocytes and fibroblasts were also identified by the same expert. We can now qualitatively compare the "ground truth" (i.e., the expert classification) with that of the automated classifier.

- In the full image for the top-scoring well, select the "Nuclei" menu item, and select "None" for the color.
- Select the "Hepatocyte" menu item and select "Blue" for the color.
- Select the "Fibroblast" menu item and select "Red" for the color.
- Save the training set you've created by select the "File" menu item in Classifier, then selecting "Save training set...".
- Accept the default file name of "MyTrainingSet.txt" and save it to the same file as the properties file.
- Close the Classifier tool, and then close CPA.

## Activity #3: Add cellular measurements in order to improve classification

You may have noticed that while the supervised training approach does reasonably well at distinguishing the hepatocytes from the fibroblasts, upon comparing our classification to the ground truth, there are still some misclassifications. Since the number of hepatocytes is the quantity we are interested in for the screen, and they are relatively rare as compared to the fibroblasts, we want to make sure we detect them as accurately as possible.

Our approach thus far has been take a fairly unbiased approach to using image-based features: Measure as many features as we can and let the computer figure out which ones are relevant. But perhaps we can do better by noting a simple visual observation that you may have noticed from Figure 4: hepatocyte nuclei have a uniform texture, whereas the fibroblast nuclei usually contain prominent nucleoli. So we will revise our pipeline by adding a specific feature, namely, the number of speckles in each nucleus, and see if the classification improves accordingly (see the Logan and Carpenter 2010 paper listed in the "Additional Resources" section for more details on this type of approach).

Our modifications to the pipeline will do the following:

- Enhance the DNA image in order to make the nucleoli appear as bright as possible while making the rest of the DNA signal as dim as possible.
- Identify the nucleoli as individual objects from the enhanced DNA image.
- Assign each nucleoli to an enclosing nucleus and count how many nucleoli there are for each nucleus.

To do this, you are going to add three new modules to your image analysis pipeline, one for each step described above.

- Select the *MeasureObjectSizeShape* module. We are going to be adding the new modules below this one.
- Click on the \_\_\_ button located on the bottom left of the interface. A window titled "Add modules" will appear so you can add your first analysis module to the new pipeline.
- Under "Module categories" select "Image Processing" and then select the module EnhanceOrSuppressFeatures from the list on the right-hand side. Click the "+ Add to Pipeline" button to add this module to the pipeline.
  - Please note that the "Add modules" window may obscure the pipeline panel and prevent you
    from seeing the module just added. If this is the case, the "Add modules" window can be
    moved to a different area of the screen with the mouse.
- Under the module category "Object Processing", select the module IdentifyPrimaryObjects. Click the "+ Add to Pipeline" button to add this module to the pipeline.
  - This module identifies objects in an image by detecting the foreground and then separating touching regions.
- Under the module category "Object Processing", select the module RelateObjects, which is located under the module category "Object Processing". Click the "+ Add to Pipeline" button to add this module to the pipeline.
  - This module creates an association (or "relationship") between smaller objects (called "children") enclosed by a larger object (called "parents").
  - A measurement produced by this module is the number of child objects there are for each parent object.
- Close the "Add modules" window.

### Description of the pipeline modules and their function

You can read a description of each module's function in the pipeline below and look at the corresponding module display window when you run the pipeline.

- EnhanceOrSuppressFeatures: This module processes the image in order to enhance features smaller than a scale that you select.
- *IdentifyPrimaryObjects:* This module identifies objects in an image by detecting the foreground and then separating touching regions.
- RelateObjects: This module creates an association (or "relationship") between smaller objects (called "children") enclosed by a larger object (called "parents"). A measurement produced by this module is the number of child objects there are for each parent object.
- For the EnhanceOrSuppressFeatures module, adjust these settings (leave the remaining settings at their defaults):
  - For the "Select the input image" module setting, select "DNA" from the drop-down list.
  - For the "Name the output image" setting, select "EnhancedDNA" from the drop-down list.
- For the *IdentifyPrimaryObjects*, adjust the settings for the following items (leave the remaining settings at their defaults):
  - "Select the input image": Select "EnhancedDNA" from the drop-down list.
  - "Name the primary objects to be identified": Enter "Nucleoli" as a descriptive name, which we will refer to them as, in later steps.
  - "Typical diameter of objects, in pixel units (Min,Max)": Enter "2" in the first text box, and "10" in the second text box.
  - "Discard objects touching the border of the image?" Select "No".
  - "Threshold strategy": Select "Per object" from the drop-down list.
  - o "Thresholding method": Select "RobustBackground" from the drop-down list.
  - "Masking objects": Select "Nuclei" from the drop-down list.
  - "Threshold correction factor": Enter "5" in the text box.
- For the *RelateObjects* module, adjust the settings for the following items (leave the remaining settings at their defaults):
  - For the "Select the input child objects" module setting, select "Nucleoli" from the drop-down list.
  - For the "Name the input parent objects" setting, select "Nuclei" from the drop-down list.

Now we need to see how well these settings perform. Let's use CellProfiler's "Test mode," which will allow you to see the results of your chosen settings, and adjust them as needed.

Click the "Start Test Mode" button at the bottom-left of the CellProfiler interface. You will see a slider and II icons appear next to the modules in the pipeline, as well as new buttons appear below the modules. You can now run the pipeline for a single image, either to the end of the pipeline or by telling it to stop ("pause") at a module of your choice.

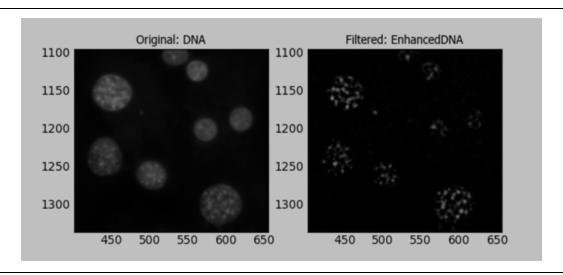
- Click the icon next to the first *OverlayOutlines* modules, which should be below the 
  RelateObjects module. When you click it, it will turn yellow ■. This will "pause" the pipeline as it runs in Test mode.
- Click on the "Run" button below the pipeline panel, in order to progress through each module in the pipeline, one by one, until it comes to the "paused" module.
- Upon stepping through the *EnhanceOrSupressFeatures* module, a module display window will appear similar to that shown at the top of Figure 9 below.
  - If the display window is hidden, double-click the module name in the pipeline to bring it to the front of the screen.

There are some image tools on the top toolbar that may be helpful to see the individual objects:

- The 1st icon lets you reset the view back to the original view.
- The 2nd and 3rd icons let you step backwards and forwards through any changes you made to the view.
- The 4th icon lets you change the view by moving in any direction in the display, by clicking and dragging.
- The 5th icon lets you change the view by zooming, by dragging and drawing a box to zoom in on.

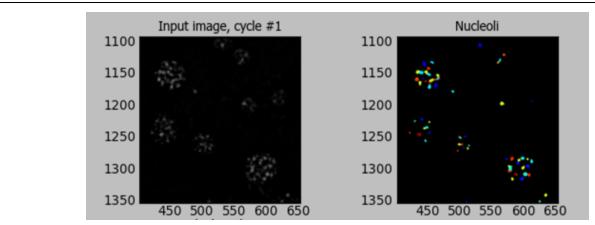
Additionally, if the image appears too dark to see much in the way of features, you can right-click on a image to bring up a context-menu to adjust the contrast to better see the image.

- From this menu, select "Image contrast", then "Log normalized" to make the dimmer pixels brighter while leaving the brightest pixels relatively unchanged in intensity.
- Zoom in and/or adjust the image contrast in order to see the quality of the image enhancement.



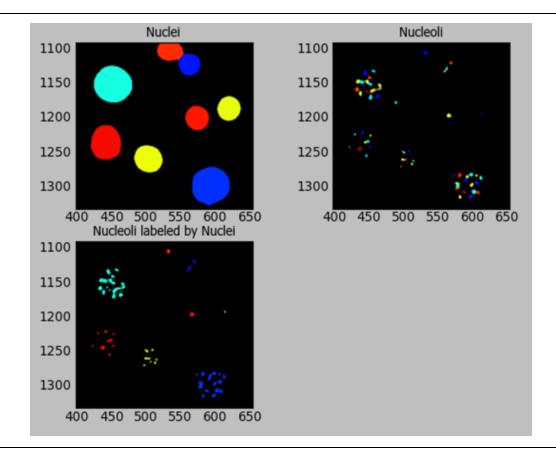
**Figure 9:** A zoomed-in view of an example module display window for the *EnhanceOrSupressFeatures* module. *Left panel:* The original raw image (pixel intensities have been log normalized for contrast). *Right panel:* The result of the image enhancement on the same section of the image.

Bring the module display window for the *IdentifyPrimaryObjects* module to the front. Again, zoom
in on the image in order to see the quality of the nucleoli identification.



**Figure 10:** A zoomed-in view of an example module display window for the *IdentifyPrimaryObjects* module. *Left panel:* The enhanced image from *EnhanceOrSuppressFeatures*. *Right panel:* The detected nucleoli found within the same section of the image, color-coded for display.

• Finally, bring the module display window for the *RelateObjects* module to the front. Zoom in on the image in order to see how the nucleoli are color-coded according to the nuclei that contain them.



**Figure 11:** A zoomed-in view of an example module display window for the *RelateObjects* module. *Top-left panel:* The identified nuclei. *Top-right panel:* The detected nucleoli produced by the *IdentifyPrimaryObjects* module found within the same section of the image, color-coded for display. *Bottom-left panel:* The nucleoli assigned to the enclosing nuclei, color-coded by the nuclei.

- Exit Test mode by clicking the "Exit Test Mode" button at the bottom-left of the CellProfiler interface.
- Click the button labeled "Analyze images" at the bottom-right of the CellProfiler interface to start the pipeline. As before, the software will identify all the nuclei in a well and make measurements on each, but this time will include the nucleoli count for each nucleus.

The pipeline will run beginning with the first of 15 images, and the window will update 15 times, each time the next image is analyzed. The full run may take a few minutes.

# Activity #4: Evaluate the improvement in classification

- Start CellProfiler Analyst by double-clicking the icon from the Start Menu (Windows) or the Applications folder (Mac).
- When CPA asks for a "properties" file, select the fille "Activity1.properties", the same file that was used before for Activity #2, located on the Desktop (or wherever you set your Default Output Folder).
- When the CPA interface appears, select the Classifier icon in the CPA menu ( , 1st on left).

- When the Classifier interface appears, select the File menu item, and select "Load training set...".
   Select the file "MyTrainingSet.txt" which you created at the end of Activity #2.
  - This will load the positive and negative nuclei you selected for the training set earlier. Since we
    are using the same nuclei as before, there's no need to create another training set.
- Click the "Train classifier" button.

#### Question 4

• After pressing the "Train classifier" button, what is the top-most image feature now? Did it change from what was before?

Most likely, the top-most rule (possibly the only rule) is of a form similar to "IF (Nuclei\_Children\_Nucleoli\_Count > 0.0,...". What this rule essentially says is the following:

- o If a nucleus has more than zero nucleoli, the nucleus is classified as a hepatocyte.
- o If a nucleus has no nucleoli, the nucleus is classified as a fibroblast.

To confirm the performance of this rule, we will look at a full image as an example, and then score the screen.

- Double-click any of nuclei thumbnails in the positive or negative bins.
- From the image that opens, click "Classify" from the menu, then "Classify Image".
- The nuclei will be color-coded according to their classification based on the current rules.
- Select the "Nuclei" menu item, and select "None" for the color.
- Select the "Hepatocyte" menu item and select "Blue" for the color.
- Select the "Fibroblast" menu item and select "Red" for the color.

### **Question 5**

- As you look at the full image, do you see any misclassifications based on your training set versus that based on the ground truth? If so, how many? Is it more or less than before?
- Press the "Score all" button. A dialog box will appear with scoring options; click "OK" to accept the
  default settings and begin scoring. Every cell in every image will now be scored as positive or
  negative by the new classifier you built.

A "Hit table" window will appear containing the summarized scores for every image. The total cell count is reported, as well as the number of positive and negative cells classified, with the last column is the enrichment score.

- Click on the column header labeled "Enriched Score positive." (You can resize the hit table window, if this column is not visible). Clicking this header will sort the rows in ascending or descending order, according to the enrichment scores. Sort the column values so the order is descending, with the highest score at top.
- Double-click on the asterisk in the first row to the left of the first column ("ImageNumber") to display the corresponding full image for the top-scoring well.
- From the image that opens, click "Classify" from the menu, then "Classify Image" to see the final classification.

## **Conclusions**

Drug discovery requires a means to determine chemical toxicity as early as possible in the development process. Growing hepatocytes co-cultured with supporting fibroblasts provides an excellent solution to this problem, but the challenge remains of producing enough hepatocytes under these conditions for full-fledged drug research. We therefore screened co-cultures for chemicals that induce hepatocyte proliferation, but due to biological considerations, we are limited to only one fluorescence stain to distinguish the two cellular phenotypes, with the further challenge of performing this task upon thousands of experimental conditions.

To accomplish this goal, we used an automated approach by first collecting a wide array of cellular image features from an actual high-content experiment, and then training a machine learning classifier to distinguish the two phenotypes based on these features. We then improvised the results of this classification by intelligently collecting additional image features based on our observation of the phenotypes, allowing us to then scoring several thousand cells, opening the door to scaling up this assay to millions of images. By using open-source software specially designed to allow biologists to analyze their images, we were able to accomplish these tasks, all without the need to earn a computer science degree.

## **Software Used In This Activity**

CellProfiler and CellProfiler Analyst

Developed by Anne E. Carpenter, Thouis R. Jones and Lee Kamentsky at the Broad Institute of Harvard and MIT. Available for free download at: <a href="http://cellprofiler.org">http://cellprofiler.org</a>

#### **Related Web Sites**

CellProfiler: Cell Image Analysis Software: http://cellprofiler.org

#### <u>Additional Resources</u>

Shan J, Schwartz RE, Ross NT, Logan DJ, Thomas D, Duncan SA, North TE, Goessling W, Carpenter AE, Bhatia SN. Identification of small molecules for human hepatocyte expansion and iPS differentiation. Nat Chem Biol. 2013. 9(8):514-20. (Pub Med ID: 23728495)

Khetani SR, Bhatia SN. Microscale culture of human liver cells for drug development. Nat Biotechnol. 2008. 26(1):120-6. (Pub Med ID: 18026090)

Soldatow VY, Lecluyse EL, Griffith LG, Rusyn I. *In vitro* models for liver toxicity testing. Toxicol Res (Camb). 2013 Jan 1;2(1):23-39. . (Pub Med ID: 23495363)

Logan DJ, Carpenter AE. Screening cellular feature measurements for image-based assay development. J Biomol Screen. 2010 Aug;15(7):840-6. (Pub Med ID: 20516293)