

# Parameter optimization with Paramorama

parameter visualization for image analysis

## Introduction

Image analysis algorithms are often highly parameterized and much human input is required to optimize parameter settings.

**Paramorama** makes this easier by enabling the user to interactively explore image analysis results in the context of sampled parameter space. It offers:

- Different views on the data;
- Interactive capabilities to query and filter the data;
- Tagging of results; and
- Exporting of regions of interest for further analysis.

It can be downloaded at: [www.comp.leeds.ac.uk/scsajp/applications/paramorama/](http://www.comp.leeds.ac.uk/scsajp/applications/paramorama/)

We also developed a sampling plug-in for **CellProfiler**. It enables the user to automatically sample parameter values, compute corresponding results and save these to disk. The plug-in ships with the developer's version of CellProfiler: [www.cellprofiler.org/developers.shtml](http://www.cellprofiler.org/developers.shtml)

## Overview of Paramorama

**Paramorama** has coordinated views; interaction in one updates others.

The screenshot shows the Paramorama interface with five main components:

- Metrics**: A histogram view showing the distribution of detected objects for different parameter values.
- Parameter space**: A hierarchical tree view of sampled parameter space, showing how parameters like 'Threshold' and 'Object Size' are nested.
- Preview area**: A grid of small thumbnail images showing the results of different parameter settings.
- Reference images**: A list of input images with their corresponding processed versions shown side-by-side.
- Widget panel**: A panel on the right containing filters and settings for the preview area.

Blue arrows indicate interactions between the views: dragging a box in the Metrics view highlights the same region in the Parameter space view; clicking on a thumbnail in the Preview area highlights the corresponding reference image in the Reference images view; and the Widget panel controls both the Preview area and the Parameter space view.

## 1. Sample parameters in CellProfiler

The screenshot shows the CellProfiler pipeline editor with two open dialog boxes:

- Sampling dialog**: Shows settings for 'Typical character of objects in primary (Intensity)'. It includes fields for 'Lower bound on the character range' (0.0), 'Upper bound on the character range' (1.0), and 'Number of samples' (10).
- Sampling settings dialog**: Shows a table for 'Sampling settings for module Identity.Primary.Objects'. It lists parameters: 'Typical character of objects in primary (Intensity)' with current value 0.5, lower bound 0.0, upper bound 1.0, and number of samples 10; and 'Threshold factor' with current value 1.0, lower bound 0.0, upper bound 1.0, and number of samples 10.

A large blue arrow points from the top of the pipeline editor towards the sampling dialogs.

### Initialize sampling

1. The user selects a module in the pipeline view.
2. The user initializes sampling from the menu bar by selecting: **Sample > Initialize sampling**

### Specify sampling settings

The plug-in automatically detects and lists all continuous parameters for the module currently selected in the pipeline view.

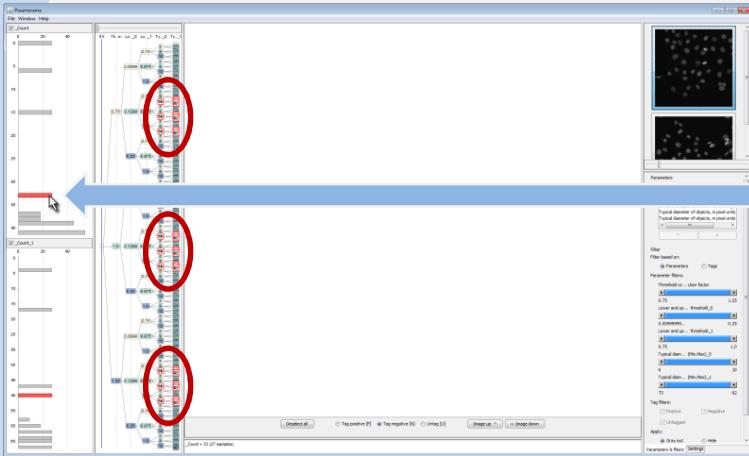
3. The user selects a subset of the parameters, specifies an interval for each, and specifies how many samples to compute for each interval.
4. To start sampling, the user clicks the **Sample** button.

The plug-in samples each interval uniformly, computes all unique combinations of values, and feeds these to CellProfiler. Parameter values and image-based results are saved to disk.

# Parameter optimization with Paramorama

parameter visualization for image analysis

## 2. Load & analyze results in Paramorama



### Load data

1. The user loads a first dataset by selecting: **File > Open data file**
2. The user adds a second dataset by selecting: **File > Add data**
3. The user adds a reference image for each dataset by selecting: **File > Add reference image**

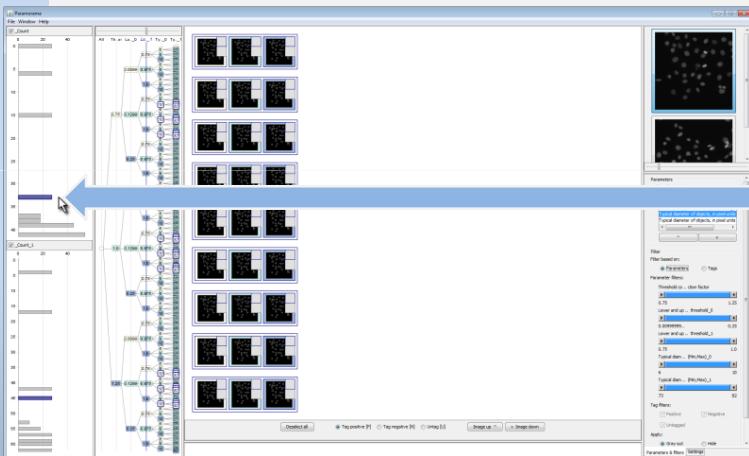
### Brushing

The two metric views show histograms of the number of objects detected in the two reference images.

4. The user moves the mouse cursor over bars in the top metric view. The bar in focus is highlighted in red. The results brushed in this way are also highlighted in red in the second metric view and in the parameter space view.

Note the regular distribution of results in parameter space.

## 3. Investigate area of interest



### Selection

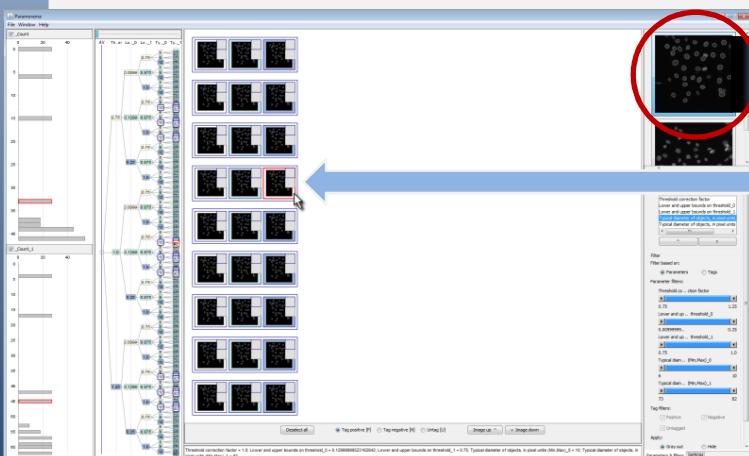
The user decides to investigate the results highlighted in the previous step in more detail.

- The user selects the highlighted bar in the top metric view by clicking it.

Selections are highlighted in dark blue. All regions in parameter space that correspond to this selection are also selected.

The results that match the selected regions of parameter space are shown as thumbnails in the preview area. The top-to-bottom order of the thumbnails match the top-to-bottom order of selections in the parameter space view.

## 4. Review results



### Brushing

1. The user moves the mouse cursor over some of the thumbnail images.

The thumbnail under the cursor is highlighted in red and overlaid on the reference image at the top right.

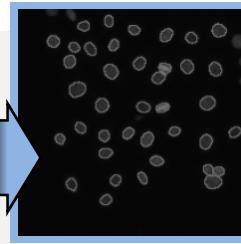
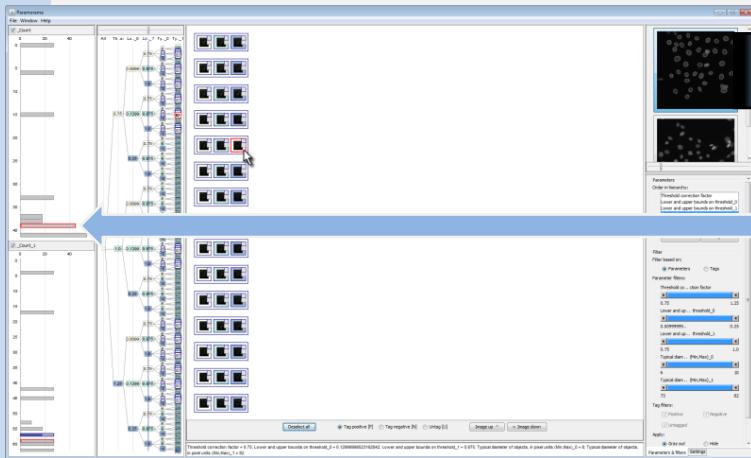
2. By inspecting the image-based results, the user sees that some objects are not accurately detected.

Note, in the magnified image above, that not all cells are outlined. The selected object count in the first metric view is too low.

# Parameter optimization with Paramorama

parameter visualization for image analysis

## 5. Change area of interest



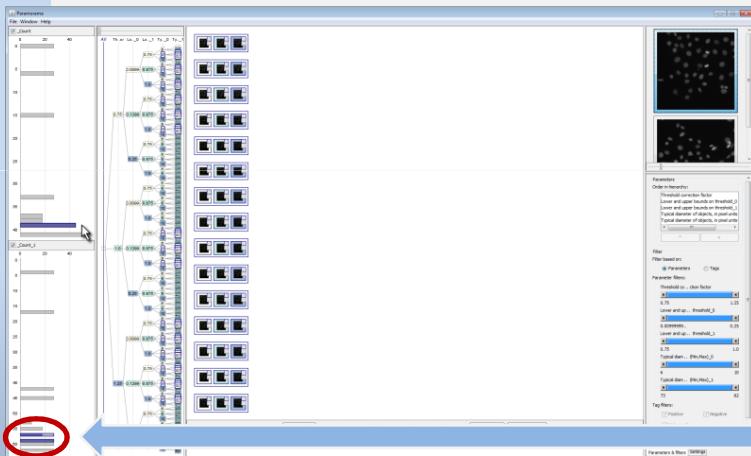
1. The user now selects a bar corresponding to a higher object count in the first metric view.

As before, the selection is highlighted in other views including the second metric view and the parameter space view. Thumbnails of the corresponding image-based results are shown in the preview area.

2. The user brushes some of the thumbnails in the preview area and inspects the details in the reference image view.

The user is now happy with the quality of the object detection for the first data set.

## 6. Investigate findings across datasets



Satisfied with the selected results for the first dataset, the user decides to investigate the corresponding results for the second data set.

The user notices that the selection in the first metric view is distributed across two bars in the second view. This means that the parameter settings that yield a single object count for the first data set, yield two distinct object counts for the second data set.

The user decides to investigate the two object count results for the second data set separately.

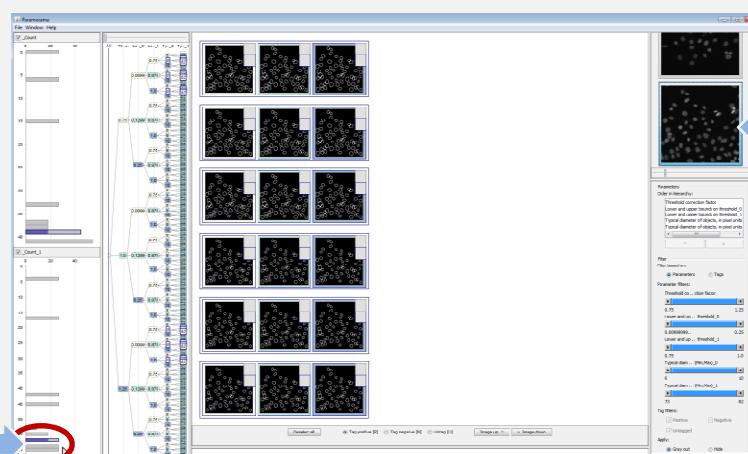
## 7. Refine selection

### Deselection

1. The user deselects the second bar in the second metric view by clicking on it.

The second bar is no longer highlighted and all results with a corresponding object counts are deselected in other views.

Note that fewer thumbnails are displayed in the preview area.



### Change dataset

2. The user selects the second dataset by clicking on the second reference image in the reference image view.

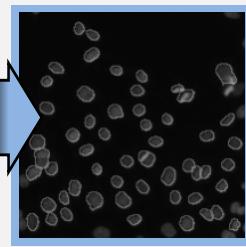
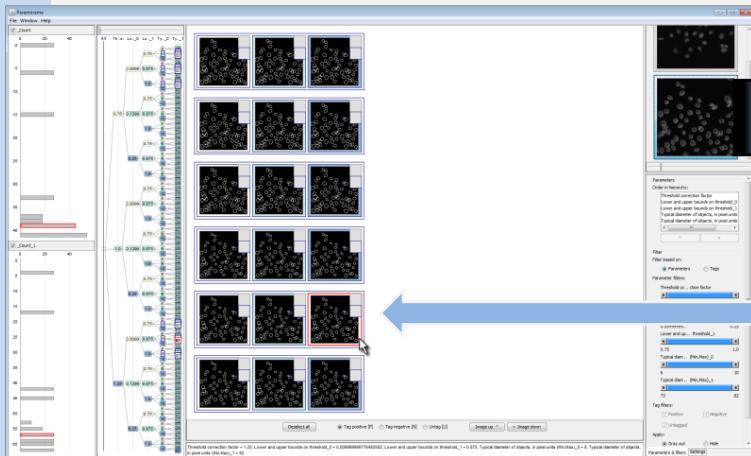
The second image is highlighted in reference image view meaning that the second dataset is active. All thumbnails in the preview area are updated to show image-based results for the second dataset.



# Parameter optimization with Paramorama

parameter visualization for image analysis

## 8. Review results

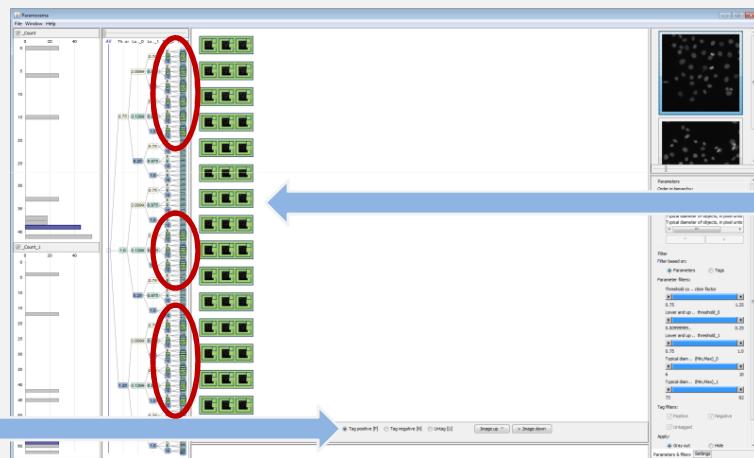


1. The user checks the selected results by brushing the thumbnails in the preview area and is happy with the quality of the object detection.
2. The user repeats the previous two steps, but now selects the second bar in the second metric view (see Step 6). The user is also satisfied with the quality of these results.

## 9. Tag good results & analyze parameters

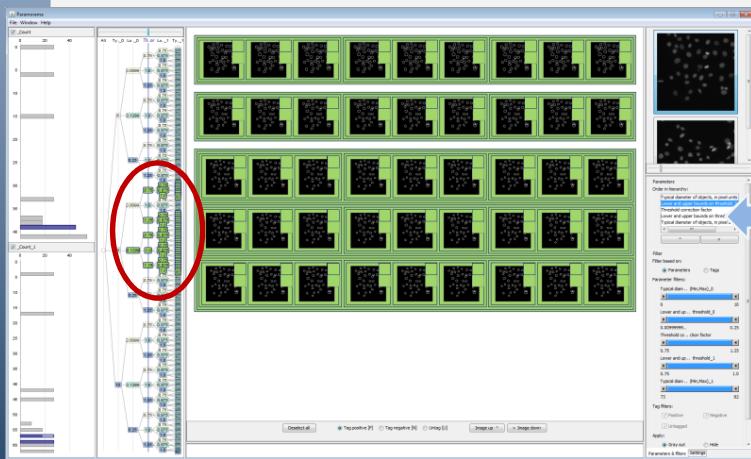
**Tagging**  
The user wants to capture their findings and decides to tag the results.

1. The user first makes sure that the tag mode is “positive” by selecting **Tag positive** in the tool tray below the preview area.



2. The user now clicks on the results in the preview area.  
The thumbnails are shown with a green background in the preview area. The regions in parameter space that contain the results are also drawn in green.  
The user sees that there is a regular pattern of positively tagged regions in the parameter space view.

## 10. Reorder parameters



### Change parameter order

After spotting the pattern in parameter space, the user decides to reorder the parameters to find the contiguous region of parameter space that yields good results.

- The user changes the order of two parameters in the parameter space view by interacting with **Parameters > Order in hierarchy** in the widget panel.

The hierarchy in the parameter space and the layout in the preview area are updated.

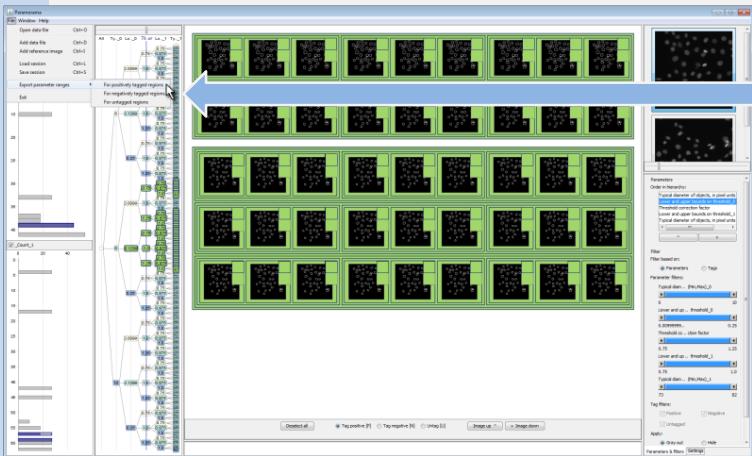
The user has isolated a contiguous region in parameter space that contains all the positively tagged results.



# Parameter optimization with Paramorama

parameter visualization for image analysis

## 11. Export tagged results



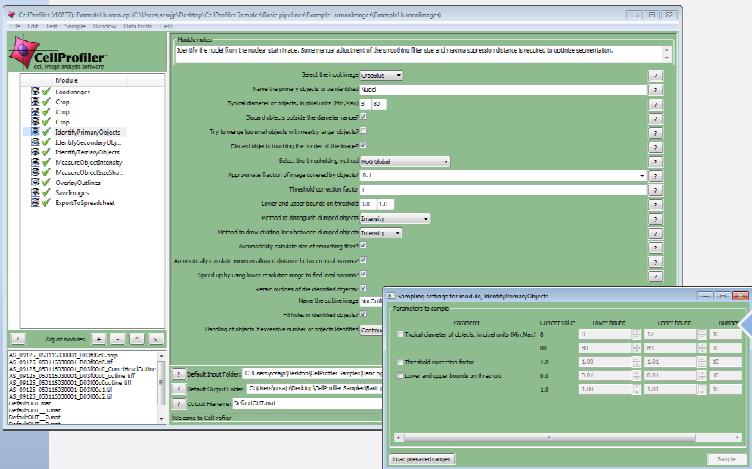
### Export tagged regions

The user wants to sample the region tagged as positive more finely. To do this, the user exports the region tagged as positive.

1. The user selects  
**File > Export parameter ranges > For positively tagged regions**

The positively tagged region is saved to disk as an XML file that can be read by the CellProfiler sampling plug-in.

## 12. Import ranges in CellProfiler & resample



### Import ranges and resample

The user would like to resample the region of parameter space that has been identified more finely.

1. The user clicks on the **Load presaved ranges** button
2. The user navigates to the file saved in the previous step.

The parameter ranges are automatically updated to the presaved ranges.

3. The user clicks on the **Sample** button to initialize sampling.

## Further information

Paramorama is developed and maintained by:

Dr AJ (Hannes) Pretorius, School of Computing, University of Leeds  
E: a.j.pretorius@leeds.ac.uk  
W: [www.comp.leeds.ac.uk/scsajp/](http://www.comp.leeds.ac.uk/scsajp/)

Major contributions were also made by:

Dr Roy Ruddle, School of Computing, University of Leeds  
Dr Mark Bray, Imaging Platform, Broad Institute of MIT and Harvard  
Dr Anne Carpenter, Imaging Platform, Broad Institute of MIT and Harvard  
Virtual Pathology, University of Leeds (Dr Darren Treanor, Dr Derek Magee and John Lau)

To cite our work and for further information, please see:

A.J. Pretorius, M.-A.P. Bray, A.E. Carpenter and R.A. Ruddle, *Visualization of parameter space for image analysis*, IEEE Transactions on Visualization and Computer Graphics, 2011.