Junction Mapper: user notes and algorithm definitions

Introduction

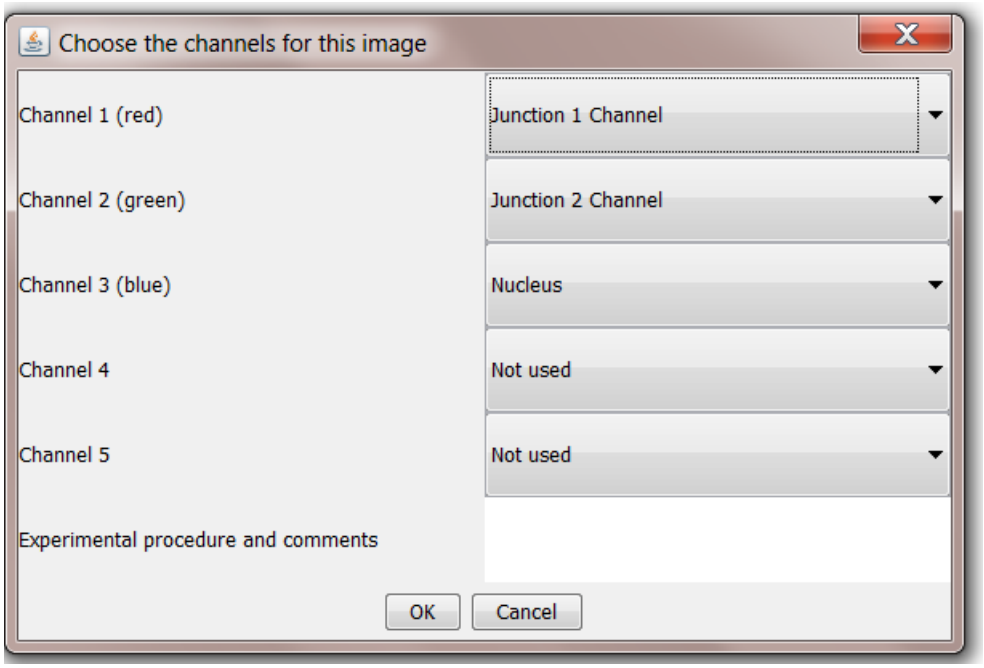
Junction Mapper is a semi-automated software application for analysing data from images of cells in close proximity to each other in monolayers. The focus of Junction Mapper is to measure the morphology of cell boundaries, define single junctions and quantify the length, area and intensity of the staining of different proteins localised at cell-cell contacts. The output are various unique parameters that assess the contacting interface between cells and up to two junctional markers. Here we describe the operational mode of the software and how the different steps and parameters are calculated computationally.

Image Loading

Junction mapper is suitable for analysis of cell images that have the following properties:

- The system will only load single images saved in the tiff format.
- The images must have three channels that correspond to the following features;
    - Cell boundaries (Junction marker 1)
    - Cell nuclei
    - A measurement channel (Junction marker 2)

Upon loading an image, the user is asked to define the channels in the tiff image where the three expected image analysis components are located (Junction 1 Channel, Junction 2 Channel and Nucleus). These can be defined in Junction Mapper for images stained with any combination of fluorescent conjugates. Users may add notes to their analysis in the text box shown in the image above. These will appear with output materials.

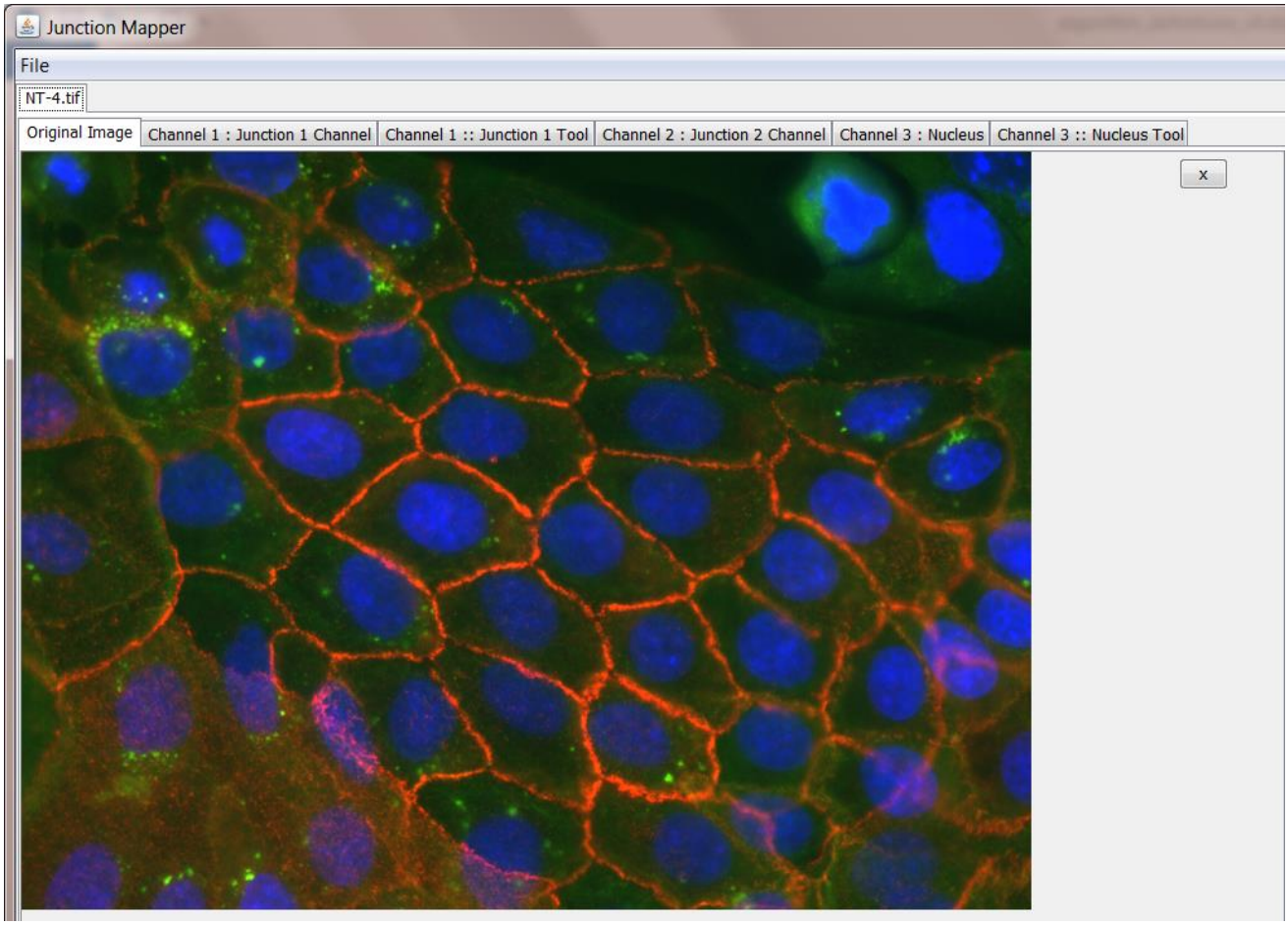


Data Output

When Junction Mapper is started by the user, an output directory for the analysis data is automatically created as a subdirectory of the directory location where Junction Mapper was started from. This directory is named in the following format <dd>_<mm>_<yyyy>_hhmm, so if junction mapper was started on 12th April 2019 at 09:18 the output directory created would be called; 2019_04_12_0918. Data is saved by Junction Mapper when the 'Save as Spreadsheet' controls are used.

General Operation

Once the cell image is loaded, the user should see the corresponding image (example below). There are six tabs, four of which contain the original images and two tool tabs (Junction 1 Tool and Nucleus Tool). These two tools operate independently of the other.



Junction 1 Tool

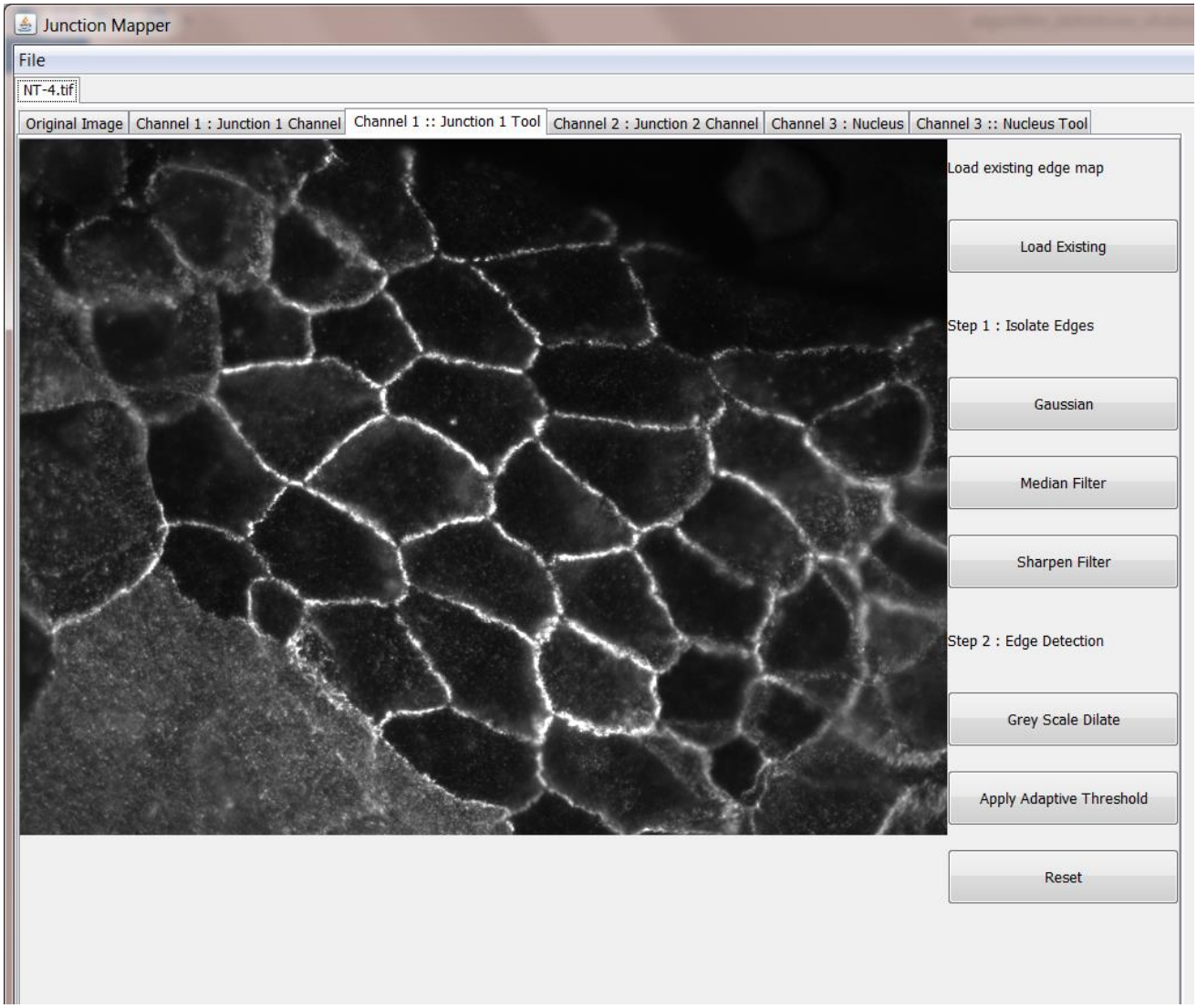
This tool is used to measure distinct parameters at the cell boundary using the Edge channel and the measurement channel of the original image. To use the membrane tool, select the appropriate tab in the image. In order to obtain the parameters, a number of steps are performed via the membrane tool and outlined below:

- Step 1: Edge Detection
- Step 2: Produce Binary Edge Map
- Step 3: Finesse the edge map
- Step 4: Select cells to be analysed
- Step 5: Select Individual Cell to Analyse
- Step 6: Define corners of cells
- Step 7: Measure different parameters at cell-cell contacts

Step 1: Isolate Edges

In order to get meaningful results from Junction Mapper, it is very important to construct an accurate representation of the cell boundary. Cell images can be taken with varied magnifications, resolution and image quality. Junction Mapper is sufficiently flexible to perform well on many different image types. The first step is to emboss the edges of the cells by using the tools on the panel. There are three filters on the right side panel:

- **Gaussian** - blurs cell edges
- **Median** - makes edges more uniform and removes spot noise
- **Sharpen** - makes edges stand out more

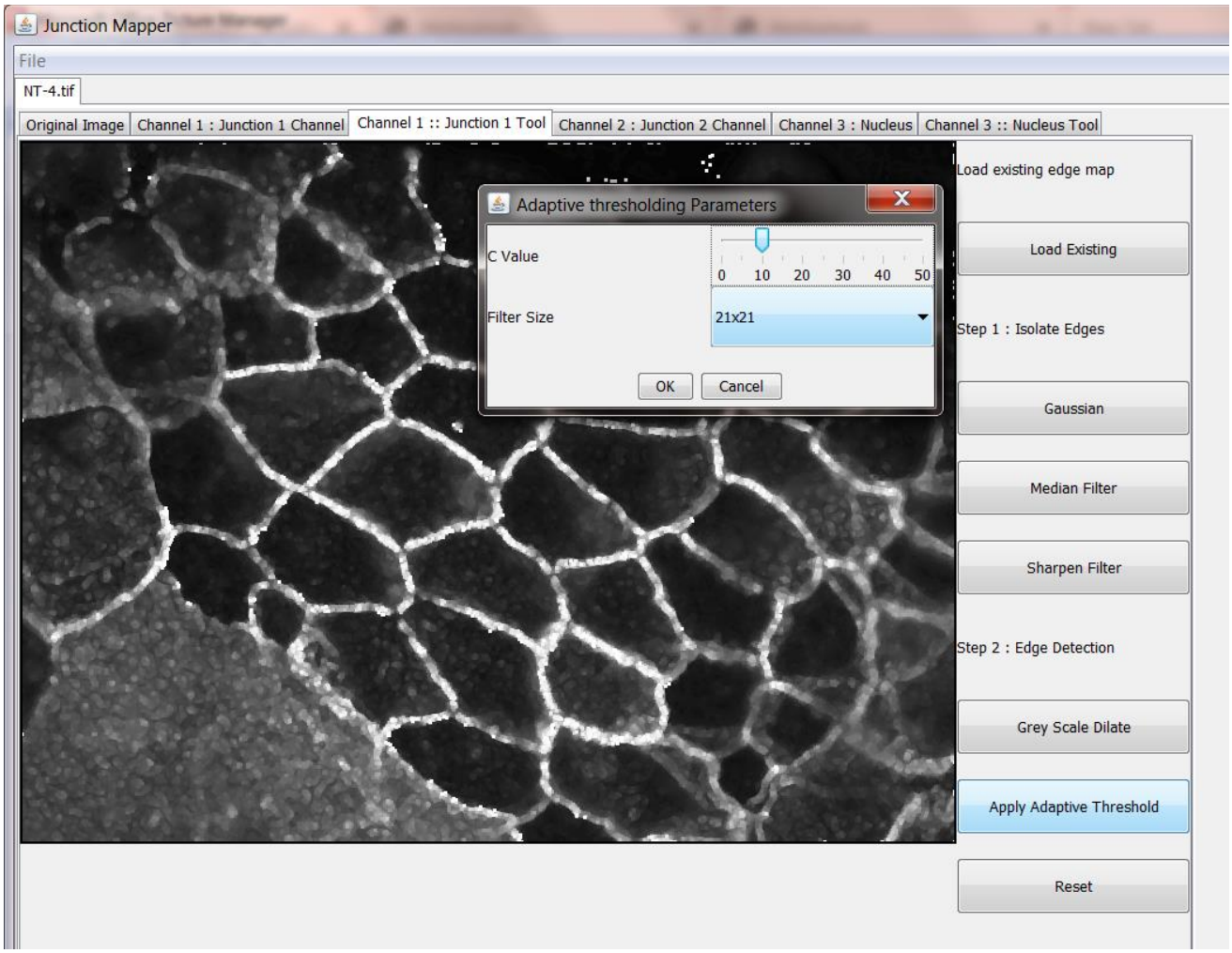


Step 2: Produce Binary Edge Map

The **Grey Scale Dilate** function can be used to fill any holes in the cell edge detected. The number of repetitions of these filters and the order and combinations that they are applied in is user-controlled and should be guided by the resultant image itself, which is displayed after every operation. Once the edges are well defined, the user should select the **Apply Adaptive Threshold,** which will binarize the image. The C value and filter size parameters to this operation are chosen from a dialog box that appears when this control is selected. The C value chosen depends on the quality and contrast of the image (higher contrast, higher C value), whilst the size of the filter selected should be slightly bigger than half of the average edge width in pixels. The result of the **Apply Adaptive Threshold** function is a binary image that is used as the basis to build the edge map. Existing edge maps saved previously can also be loaded into the system at this stage.

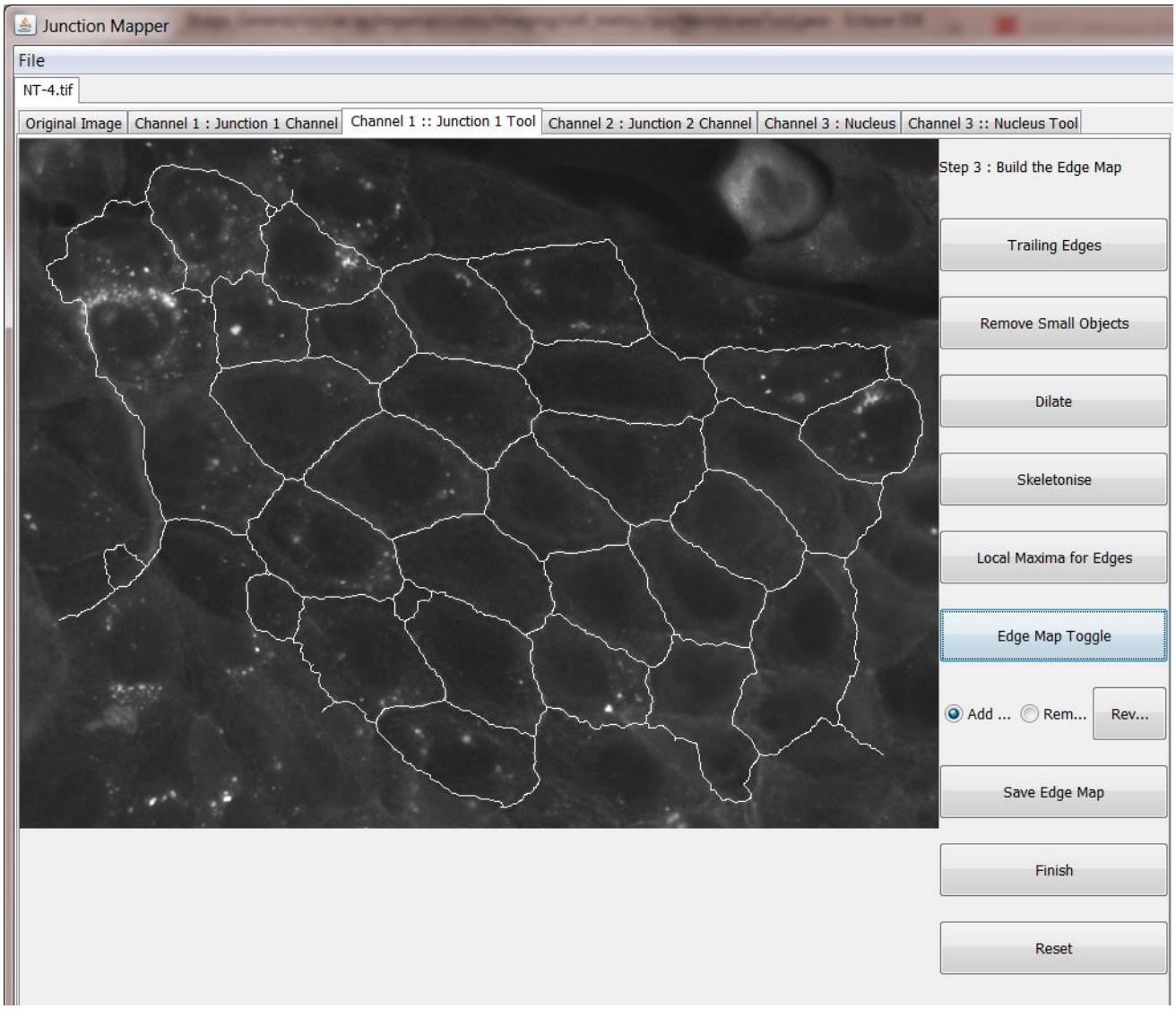
Control Name	**Algorithm Used**
Gaussian	Image is convolved with a 5x5 Gaussian kernel with values: `{{1,4,7,4,1},{4,16,26,16,4},{7,26,41,26,7},{4,16,26,16,4},{1,4,7,4,1}};`
Median Filter	Median Filter applied to image with a 5x5 kernel
Sharpen Filter	Image is convolved with a 3x3 sharpening filter, kernel has values: `{{-1,-1,-1},{-1,12,-1},{-1,-1,-1}}`

Grey Scale Dilate	Performs a grey scale dilation operation on the edge image in a 3x3 neighbourhood by replacing the target pixel with the largest grey scale value in the local neighbourhood
Adaptive Threshold	Binarizes the grey scale edge image by using an adaptive thresholding technique. The user chooses a C value (range [0:50]) and a filter size from the set: `{"3x3", "5x5", "7x7", "9x9", "11x11", "21x21", "35x35", "51x51","75x75","99x99"}` The size of the window should be large enough to contain pixels of the structure being detected and background pixels. A window of the chosen filter size then calculates the average pixel intensity in the window for every pixel in the image and adds the chosen C value to it. If the target pixel original grey scale value is equal to or exceeds this value (average window intensity value + C value), then the target pixel value in the resultant binary image is set to 1 otherwise it is set to 0.



Step 3: Finesse the Edge Map

At the beginning of this stage we have a binary image loosely corresponding to edges in the image that needs to be turned into an accurate edge map for further processing. The first step is usually to dilate the image two or three times (to join the detected edges) and then skeletonise (to create a single pixel wide edge). **Trailing Edges** and **Remove Small Objects** can be used to removed imperfections and misalignments in the binary image. The user should then check the accuracy of the edge map by using the **Edge Map Toggle** function to see how closely the map follows the edges in the original image – this function rotates the background on which the skeleton is drawn between blank, Edge channel and measurement channel allowing the user to assesss the accuracy of the skeleton produced. Minor adjustments can be made by using the **Local Maxima for Edges function** and then **Dilating** (to join edges) and **Skeletonise** the result to get a single pixel wide edge. **The Local Maxima for Edges** can be used to locate the edge closely to the grey scale values in the original grey scale Edge image. The local neighbourhood of the original Edge image is inspected around each skeleton pixel location and a one value assigned to the maxima in this window. This operation can be performed on decreasing neighbourhood sizes in conjunction with the dilate and skeletonize functions to build a more accurate edge map. Edges can also be manually added or removed by the user by selecting the **Add** or **Remove** check buttons and clicking on the image itself. Click left to add a point and then click again to add a line between the points. This can be repeated to rapidly create an edge. Click with the right-hand button to stop the edges being added. Some of the functions will only work when the binary image is displayed (dilate, local maxima). When the edge map corresponds to the edges of the image (cells to be measured) click the **Finish** button. Completed edge maps can be saved at this stage for future use (**Save Edge Map**).

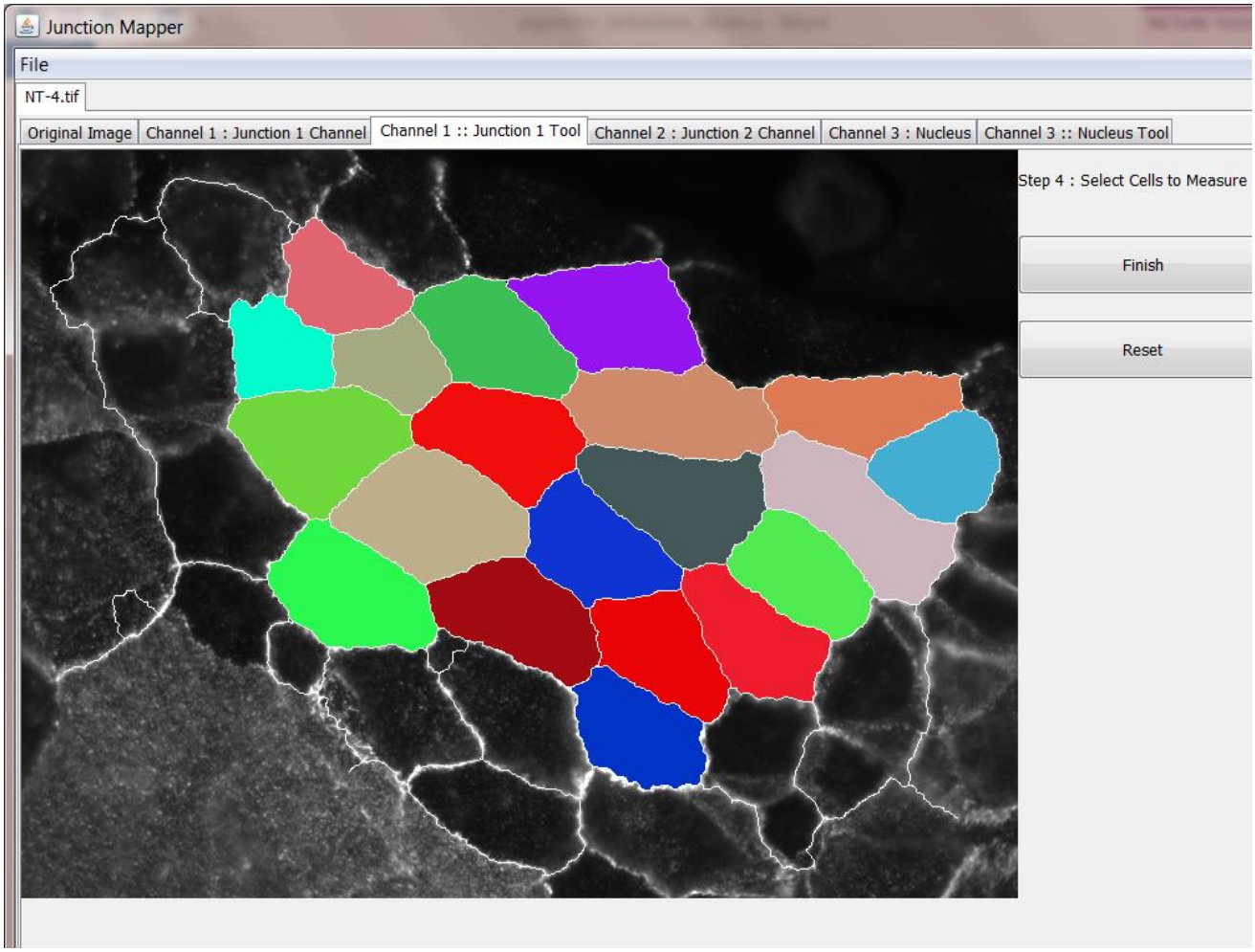


Control Name	Algorithm Used
Trailing Edges	Removes trailing edges from the image. This uses a bespoke algorithm written by the author and uses the binary edge map as input. First, the image is scanned to locate edge pixels that have only one neighbour and adds these to a list. Then, for each point in the list; 1. Remove the pixel from the target image 2. Check the neighbourhood for connected pixels, if there is only one connected neighbour add it to the list. This has the effect of removing trailing edge fragments from the edge map image.
Remove Small Objects	Removes pixel connected objects from the binary image that are smaller than a user selected threshold. User can choose from the set: `{"10","20","30","40","50","100","150","200","250","300","350","400","450","500"}` Objects smaller than the selected threshold will be removed from the image
Dilate	Standard binary image dilation algorithm
Skeletonise	Skeletonises image by using an adaption of the algorithm proposed in [1]. The masks proposed in the paper are applied in (simulated) parallel fashion to image pixels until the resultant image reaches idempotence.
Local Maxima for Edges	Alongside dilation and erosion, this technique operations is used to align the single pixel-width edges to the edges on the original grey scale image (junction marker 1) more closely. It uses the original grey scale image and the single pixel-width edge map as inputs. The user is asked to select the neighbourhood size from the set: `{"3","5","7","9","11","15","21","25"}` For each pixel in the binary edge map that is set to a 1, the neighbourhood around that pixel is inspected in the original grey scale edge channel image. The location of maximum grey scale value in the neighbourhood of junction marker 1 image is set to a one in the resultant binary image. It can be performed on gradually reducing neighbourhood sizes.
Edge Map Toggle	Rotates the background image that the binary edge map image is projected onto. Either shows just a binary image, the measurement channel or the edge channel. This control will rotate through the options in turn.
Add Straight Line	A tool to draw straight lines on the edge map image using the mouse. Select by clicking the appropriate radio button and then click left mouse button on the image at the line start point and click again with the left button at the line end point. The line will be displayed on the image. Multiple straight lines can be added by multiple mouse clicks that uses the last point as the line start point. Edges can be traced quickly using this feature. Turn off the last point by clicking the right mouse button and move to a different location.
Remove Edge	A tool to erase parts of the edge map using the mouse. Select the appropriate radio button on the interface and move the mouse over the edge fragments that you want to remove.

[1] C. Arcelli, L. Cordella, S. Levialdi, Parallel thinning of binary pictures, Electronics Letters 11(7):148 – 149, DOI: 10.1049/el:19750113

Step 4: Select cells to be analysed

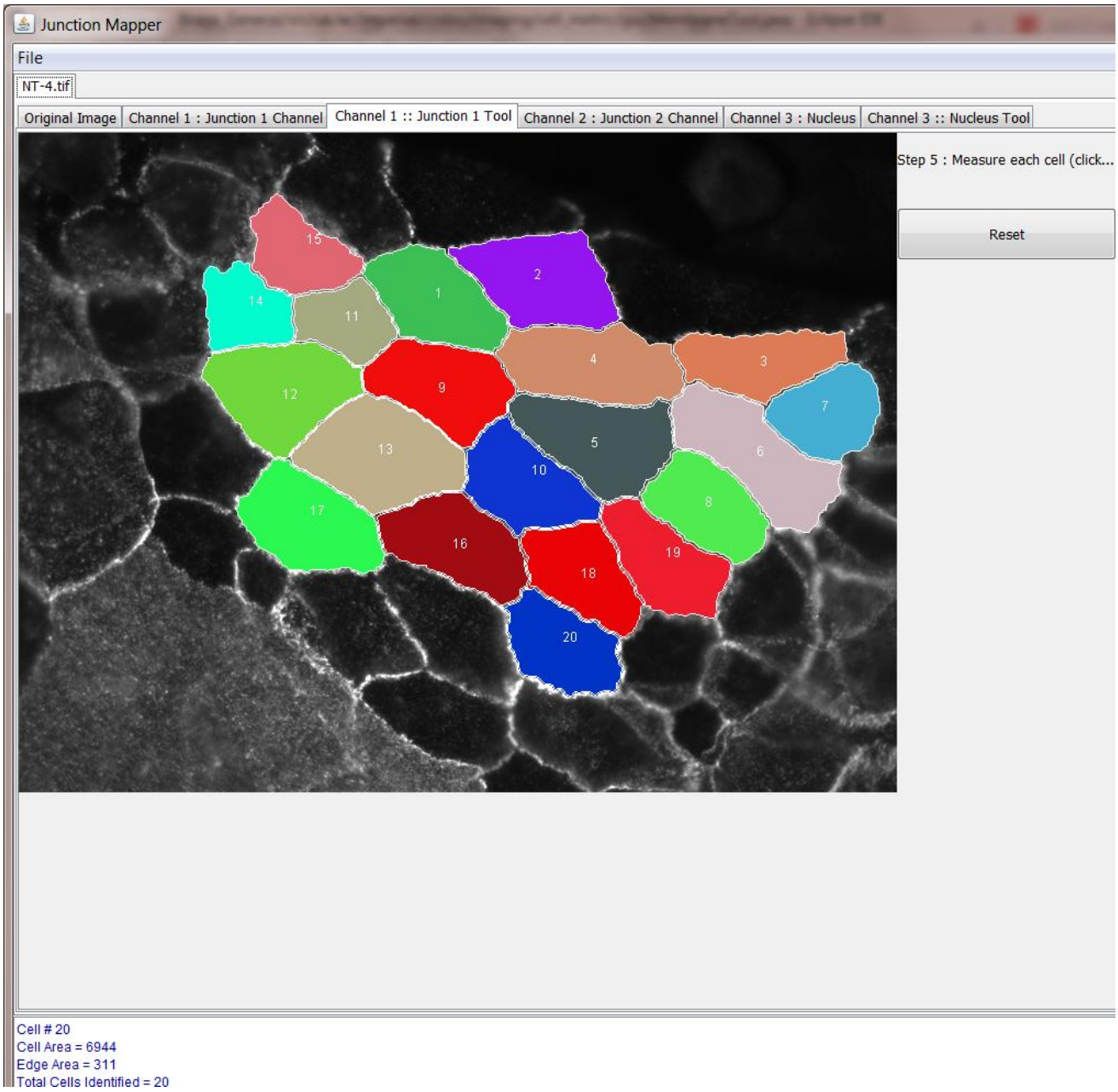
The cells to be analysed are selected in this stage by clicking inside the cell body (the cells must have closed edges for this to work). Each cell is labelled with a random colour and a unique number (in this image). At this stage, a model is made of the cell and its contour by the software. When all the cells to be analysed have been identified the **Finish** button should be pressed.



Control Name	Algorithm Used
Region Growing	Users select the cells to be measured by clicking on them with the mouse. Using the mouse click point as a seed, a region growing algorithm uses the edge map as its boundaries and identifies the pixels contained within the cell body.

Step 5: Select Individual Cell to Analyse

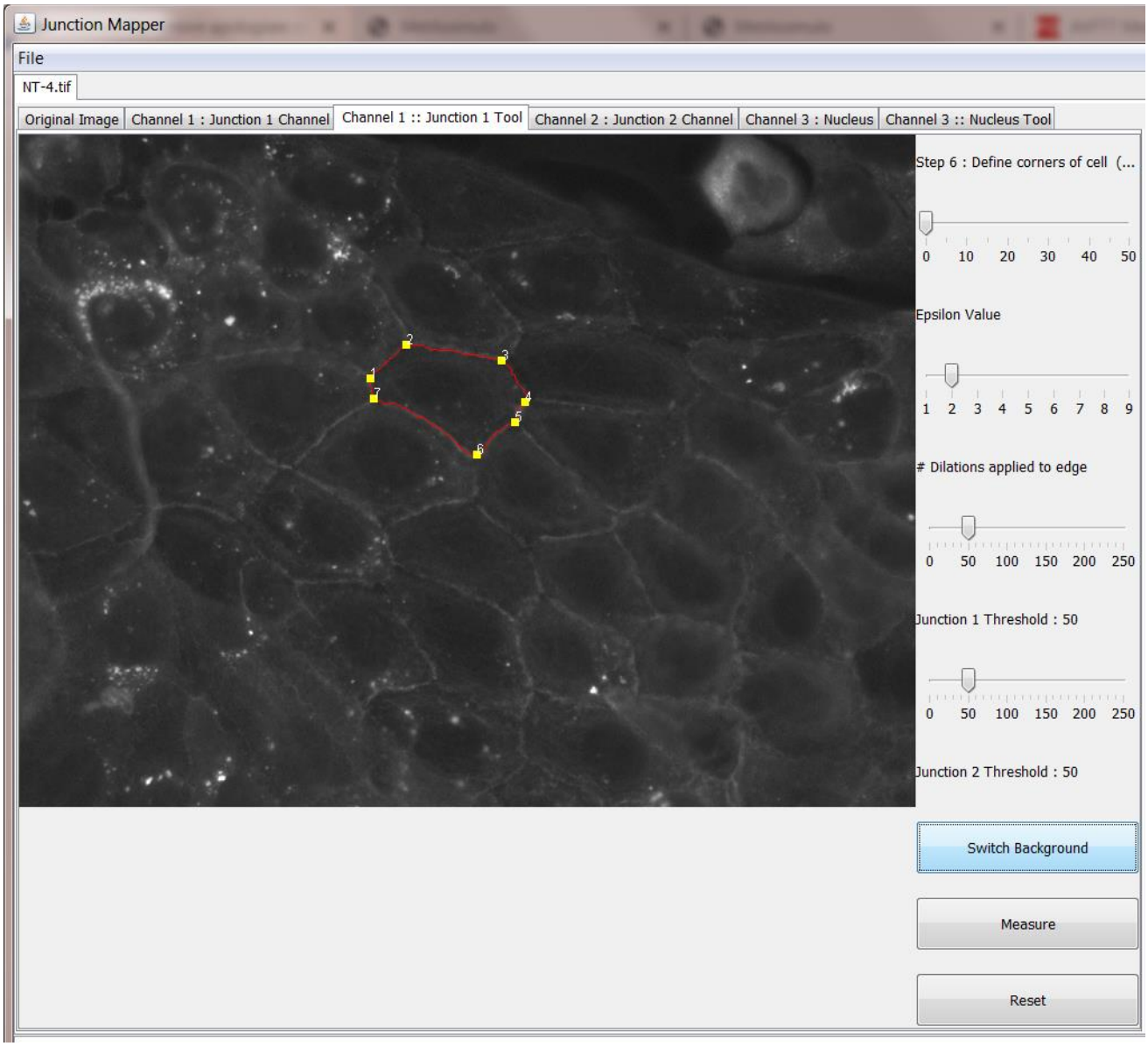
Each cell is automatically numbered. The cells are presented to the user and the user can choose which cell to analyse by clicking anywhere inside its boundary.



Control Name	Algorithm Used
Select Cell	User selects a cell to measure by clicking in its interior

Step 6: Define Corners of Cell

At this step, the corners of a cell are defined for analysis. Pulling the top slider (**Epsilon Value for Corner Detection**) to the right will cause corners to appear in suggested places on the cell boundary as numbered yellow squares. The user can keep none, all or some of these corners. Existing corners can be removed by right clicking in the yellow square, new corners can be added by left clicking on the cell boundary position. The area to measure the junction marker 1 (used to make the edge map) is altered by using the **Number of dilations** slider: this sets the area of how many dilations from the cell edge are to be measured. When this slider is changed the area of the junction marker around the edge is projected onto the image in green with the edge represented as a red contour. Global thresholds for the measurements in the edge channel Junction marker 1 image (E-cadherin Threshold below) and the Junction Marker 2 image (Measurement threshold below) can be set by the bottom two sliders. Only pixels with an intensity above the thresholds set in these images are used in the calculation of parameters for the two channels. The **Switch Background** button changes the background image onto which the binary edge map and corner image is projected for the cell. The **Measure** button quantifies the parameters for the selected cell.

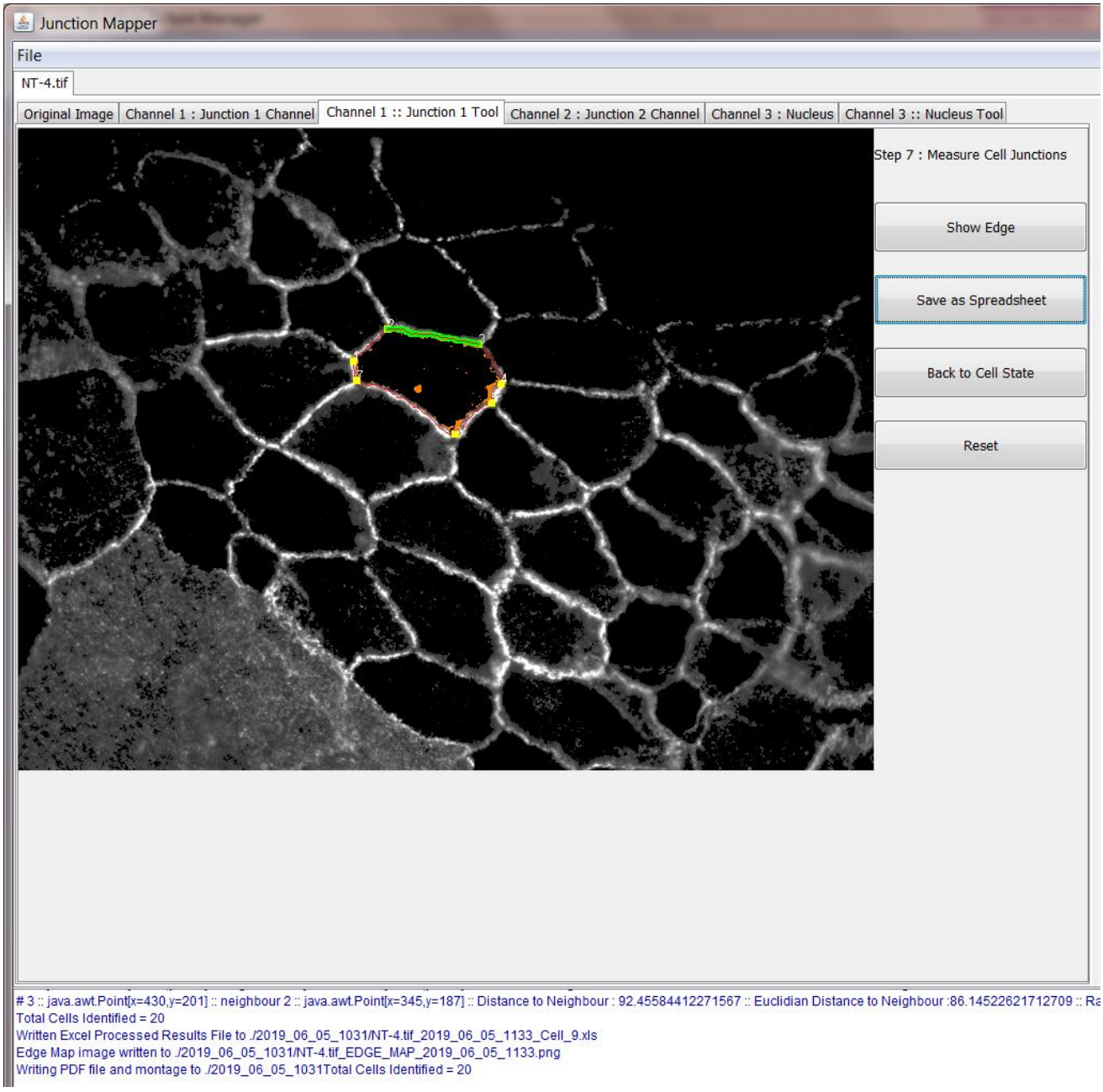


Control Name	Algorithm Used
Corner Detection	The algorithm used is based on the Douglas Peucker method defined in [2]. The user sets the epsilon value required using a slider and the corners appear in the image as defined by the algorithm. Raising the epsilon value decreases the number of corners that the algorithm adds to the image. Corners can also be manually added (left mouse click on cell edge) or removed (right mouse click on corner) from the image.
Number of Dilations applied to Edge	This refers to the number of standard image dilations applied to the single pixel cell edge and refers to the area that will be measured. The area measured of the "junction marker 1" by Junction Mapper can be altered using a slider control, the default value being 2.
junction marker 1 Threshold	A global threshold applied to the "junction marker 1" image over which pixels will be considered for inclusion in the resultant parameter calculations.
Measurement Threshold	A global threshold applied to the measurement channel over which pixels will be considered for inclusion in the resultant parameter calculations.
Switch Background	There are three options: (i) just an edge map projected on to the "junction marker 1" image, (ii) the edge map projected onto the measurement channel or (iii) the edge map plus the "junction marker 1" measurement area (the dilated edge) projected onto the "junction marker 1" channel. Using this control will rotate through the options in turn.
Measure	This button will perform the cell boundary measurements detailed elsewhere. Note that this operation does not save the measurements.

[2] David Douglas & Thomas Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", The Canadian Cartographer 10(2), 112–122 (1973) doi:10.3138/FM57-6770-U75U-7727

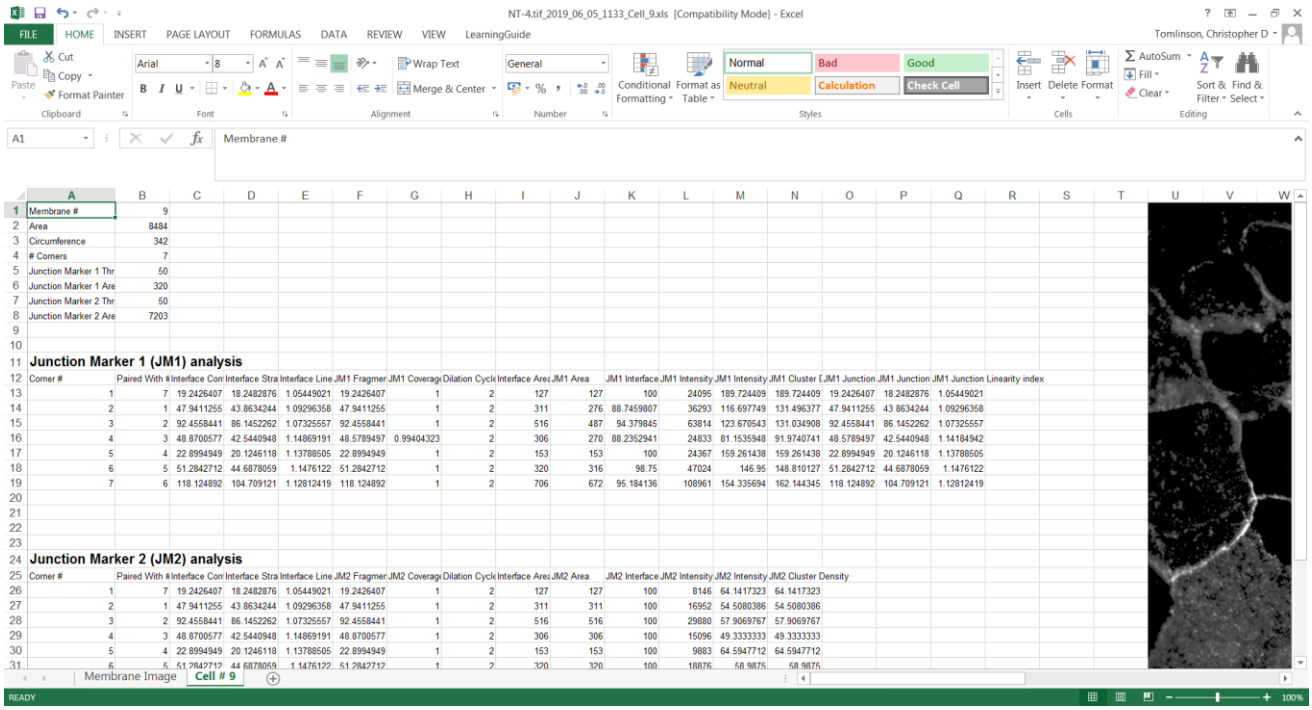
Step 7: Measure Cell Junctions

A visual representation of the measured area can be displayed edge by edge by using the **Show Edge** button. The resultant measurements and parameters calculated are saved by pressing the **Save as Spreadsheet** button. The spreadsheet file is saved in the output directory described earlier in this document. Along with the spreadsheet, several other files are saved in the output directory including image files and pdf documents containing the images produced. To return to the map of selected cells (shown in step 6), press the **Back to Cell State** button to. Another cell can then be selected for processing in the same way. This step should be repeated until all cells have been processed.



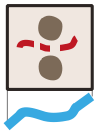
Control Name	Algorithm Used
Show edge	Shows the edges measured on the visual display individually (i.e. each contacting interface) as in the image above. Edge is displayed in red, whilst the "junction marker 1" area measured is displayed in green. Pixels in the "junction marker 1" channel that exceed the chosen threshold but are outside of the "junction marker 1" area are displayed in orange. The thresholded pixels outside the dilated area are shown as Internal junction marker 1 area or Internal junction marker 2 area at the top of the table. They are not computed in the parameters.
Save spreadsheet	Saves the calculations for the last time in a spreadsheet for the selected cell along with corresponding reference images and a pdf document
Back to Cell State	Returns to previous step so that another cell can be selected for measurement

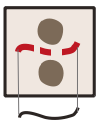
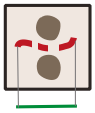
The image below shows the spreadsheet produced by the process described here. Both the Junction marker 1 image and Junction marker 2 images are analysed simultaneously. The parameters calculated for each cell-cell contact are defined in the next section.



Parameter Definitions and Formulae

This section defines the mathematical formula to calculate the primary parameters for each edge identified in a cell. Secondary parameters are calculated by normalizing values with respect of the length and area of in interface or cell-cell contacts as outlined in the Suppl Fig 3. These measurements are output to the xls spreadsheet as defined above.

#	Name	Units	Description	Mathematical Formula
1	Interface Contour 	[pixels]	Distance between two corners of the defined cell edge	$$Interface\ Contour\ = \sum_{n=cornerpixel_1}^{n<cornerpixel_2} \sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2}$$ $where$ $p(x,y)\ is\ an\ edge\ pixel; from\ an\ ordered\ list\ of\ edge\ pixels$
2	Straight-line Interface Length 	[pixels]	Straight line distance between two corner points	$$Straight\ line\ Interface\ length\ = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$ $where$ $p(x,y)_1\ is\ the\ first\ corner\ pixel\ on\ the\ edge$ $p(x,y)_2\ is\ the\ last\ corner\ pixel\ on\ the\ edge$
3	Fragmented Junction Contour 	[pixels]	Sum of stained fragments along the single pixel edge	$$Fragmented\ Junction\ Contour\ =$$ $$\sum_{n=cornerpixel_1}^{n<cornerpixel_2} if(p(x,y)_n > T\ and\ p(x,y)_{n+1} > T) \sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2}$$ $where$ $p(x,y)\ is\ an\ edge\ pixel\ from\ an\ ordered\ list\ of\ edge\ pixels$ $T\ = Threshold\ for\ staining\ intensity$
4	Dilation Cycles	[unitless]	Number of cycles used to dilate the defined edge	Number of times the binary image dilate algorithm is used to expand the defined edge. Essentially one dilate cycle changes a single pixel line to a three pixels-wide line. Two dilation cycles make the line five pixels-wide, etc.
5	Interface Area 	[pixels^2]	Area in pixels of the dilated edge area between two corners	$$Interface\ Area\ = \sum_{n=first\ dilated\ area\ pixel}^{n<=last\ dilated\ area\ pixel} p(x,y)_n$$ $where\ p(x,y)_n\ is\ a\ binary\ image\ pixel\ from\ the\ list\ of$ $dilated\ area\ pixels\ for\ this\ edge$
6	Junction marker 1 Area 	[pixels^2]	Area covered by junction marker staining within the interface area	$$Junction\ Marker\ 1\ Area\ = \sum_{n=first\ dilated\ area\ pixel}^{n<=last\ dilated\ area\ pixel} if(g(x,y)_n > T)\ \ p(x,y)_n$$ $where$ $p(x,y)_n\ is\ a\ binary\ image\ pixel\ from\ the\ list\ of$ $dilated\ area\ pixels\ for\ this\ edge$ $g(x,y)_n\ is\ the\ corresponding\ junctional\ protein\ gray\ scale\ pixel$ $T\ is\ the\ junctional\ protein\ threshold$

7	Junction marker 1 Intensity 	[A.U.]	Sum of cadherin (junctional protein) Intensity within the interface area.	$$Junction\ Marker\ 1\ Intensity = \sum_{n=first\ dilated\ area\ pixel}^{n<=last\ dilated\ area\ pixel} if(g(x,y)_n > T)\ \ g(x,y)_n$$ $where$ $g(x,y)_n\ is\ a\ junctional\ protein\ gray\ scale\ pixel$ $T\ is\ the\ junctional\ protein\ threshold$
8	Junction Contour 	[pixels]	Sum of pixel distances between the first and last junction marker pixels along the interface contour	$$Junction\ Contour\ = \sum_{n=first\ junctional\ protein\ pixel}^{n<last\ junctional\ protein\ pixel} \sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2}$$ $where$ $p(x,y)\ is\ an\ edge\ pixel\ from\ an\ ordered\ list\ of\ edge\ pixels$
9	Straight-line Junction Length 	[pixels]	Euclidian distance from first to the last pixel of junction marker 1 on the interface contour	$$Straight\ line\ Junction\ length = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$ $where$ $p(x,y)_1\ is\ the\ first\ pixel\ on\ the\ edge\ above\ the\ threshold\ for\ staining\ intensity$ $p(x,y)_2\ is\ the\ last\ pixel\ on\ the\ edge\ above\ the\ threshold\ for\ staining\ intensity$

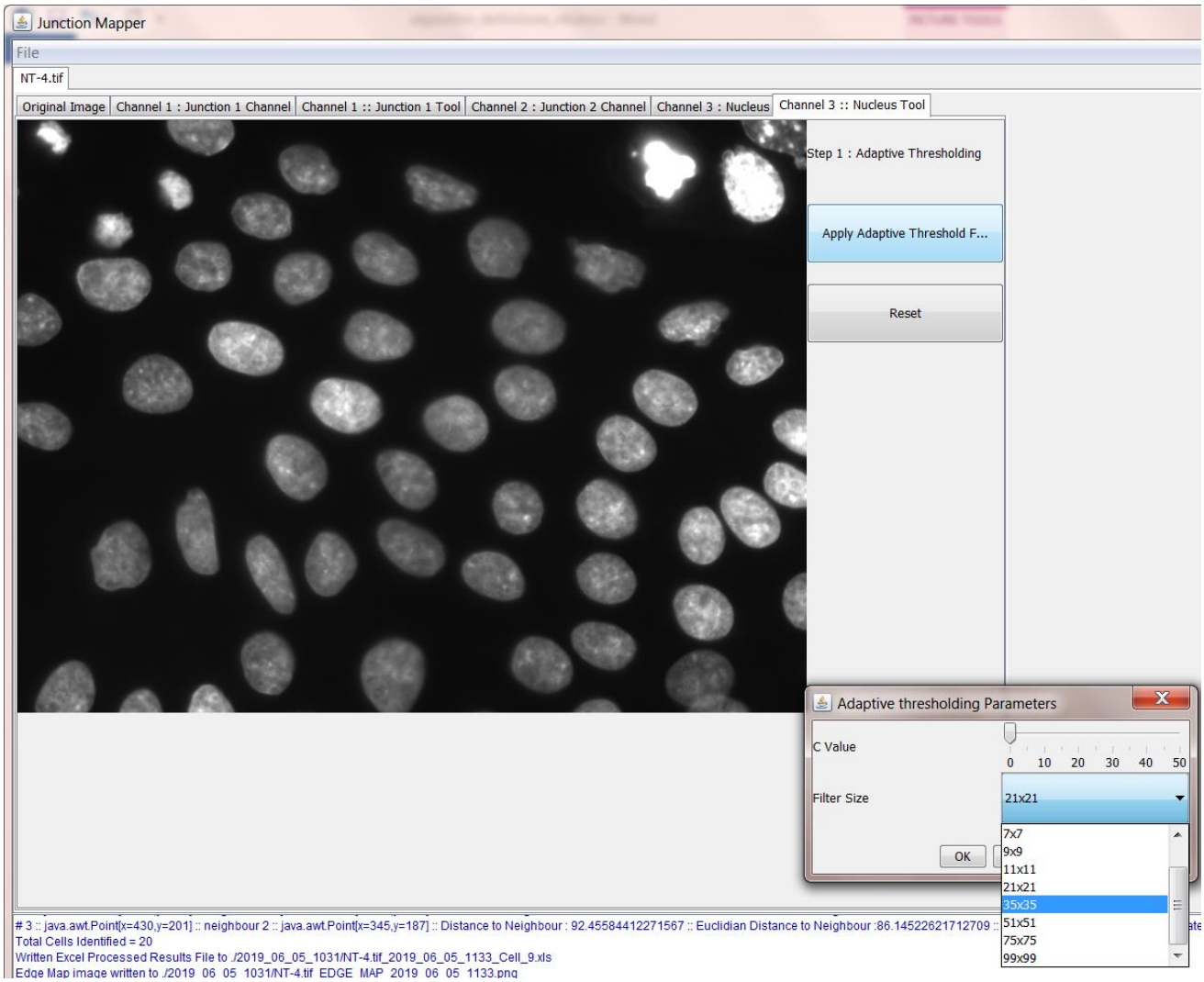
Nucleus Tool

The nucleus tool allows users to count nucleus in an image and to measure the distance between them, which can be used to infer distance between neighbouring cells. This tool is useful as an indirect measurement of the initial steps of cell scattering. Three steps are performed via the **Nucleus Tool** tab and outlined below:

- Step 1: Adaptive Thresholding
- Step 2: Tidy Nucleus Image
- Step 3: Measure Distance between Neighbouring Cells

Step 1: Adaptive Thresholding

The first step is to binarize the image via **Apply Adaptive Threshold**.

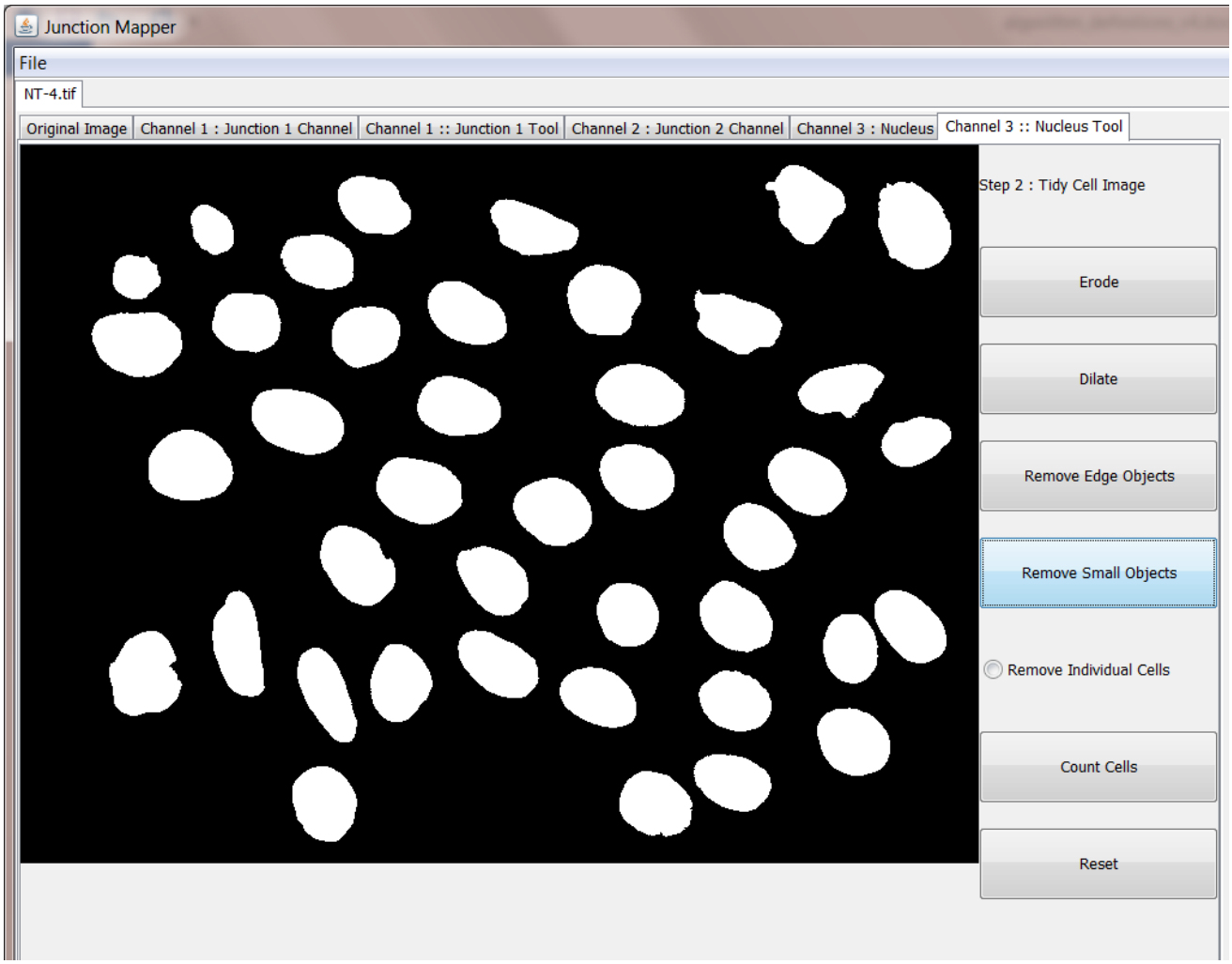


Control Name	Algorithm Used
Adaptive Threshold	Binarizes the grey scale nuclei image by using an adaptive thresholding technique. The user chooses a C value (range [0:50] and a filter size from the set: `{"3x3", "5x5", "7x7", "9x9", "11x11", "21x21", "35x35", "51x51","75x75","99x99"}` The size of the window should be large enough to contain pixels of the structure being detected and background pixels. A window of the chosen filter size then calculates the average pixel intensity in the window for every pixel in the image

	and adds the chosen C value to it. If the target pixel original grey scale value is equal to or exceeds this value (average window intensity value + C value), then the target pixel value in the resultant binary image is set to 1 otherwise it is set to 0.

Step 2: Tidy Nucleus Image

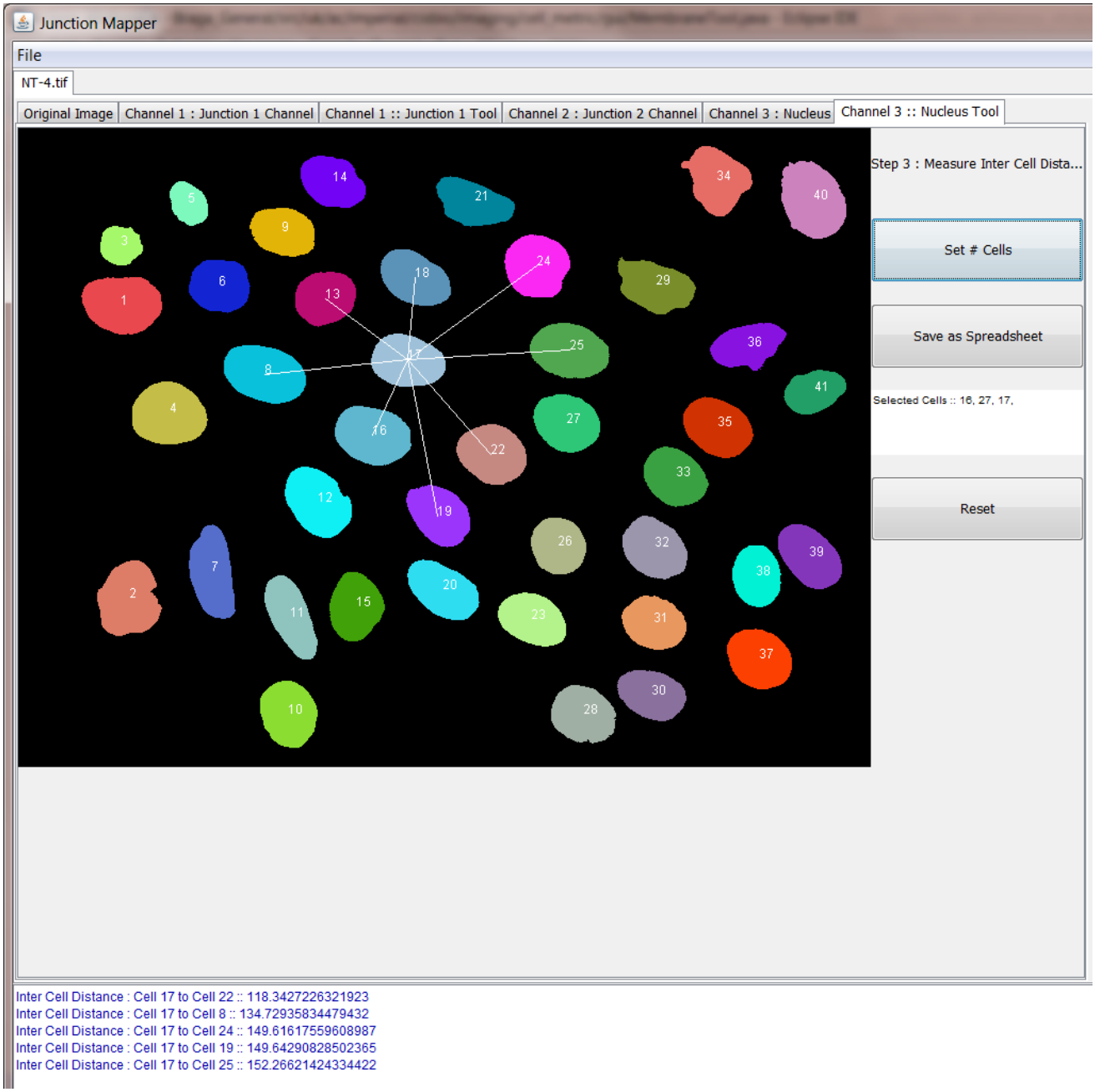
The adaptive thresholding stage create a binary image similar to the one shown below. Holes in the nuclei can be closed by dilation (**Dilate**) and cells can be returned to their original size by the erosion operation (**Erode**). Any object that touches the boundary of the image **(Remove Edge Objects)** and any small background objects = (**Remove Small Objects**) can be removed from the image by clicking on them. The **Remove Individual Cells** check button allows exclusion of selected cells by clicking on them. Finally, the remaining cells can be counted and labelled by pressing **Count Cells**.



Control Name	Algorithm Used
Erode	Standard single cycle binary image erosion.
Dilate	Standard single cycle binary image dilation. When used in conjunction with erosion can be used to remove holes inside binary objects.
Remove Edge Objects	Removes any object that touches the boundary of the image.
Remove Small Objects	Removes pixel connected objects from the binary image that are smaller than a user selected threshold. User can choose from the set: {"10","20","30","40","50","100","150","200","250","300","350","400","450","500"} Objects smaller than the selected threshold will be removed from the image
Remove Individual Cells	Removes objects from the image that the user selects by clicking on them.
Count Cells	Counts the cells in the image.

Step 3: Measure Distance between Neighbouring Nuclei

The intracell distance can be measured and output to a spreadsheet in this step – i.e. the Euclidian distance from the centre of a nucleus to the centre of neighbouring nucleus. The number of neighbouring cells that are measured can be set using the control 'Set # inter nuclei'. The cells to be measured are selected by the user by clicking on them and they appear in the list on the interface below. Cells can be removed from the list by clicking on them again. Once the cells to be measured have been selected, the distances can be saved in an Excel file (**Save as Spreadsheet**).



Control Name	Algorithm Used
Set # inter nuclei	Sets the number of nuclei from neighbouring cells that the distance will be measured for. Item needs to be selected in the pulldown menu.
Select Cell	User can select and deselect cells to be measured by clicking on them with the mouse. A list of currently selected cells is displayed on the interface.

Save as Spreadsheet	Selected nucleus measurements are calculated and output to a spreadsheet. The central point in a nucleus is calculated as; (max(x)-min(x))+min(x), (max(y)-min(y))+min(y) Where x and y are the coordinates of pixels in a nucleus. We assume that nucleus shapes are roughly symmetrical. Distances calculated are the Euclidian distance from the respective nucleus centres.