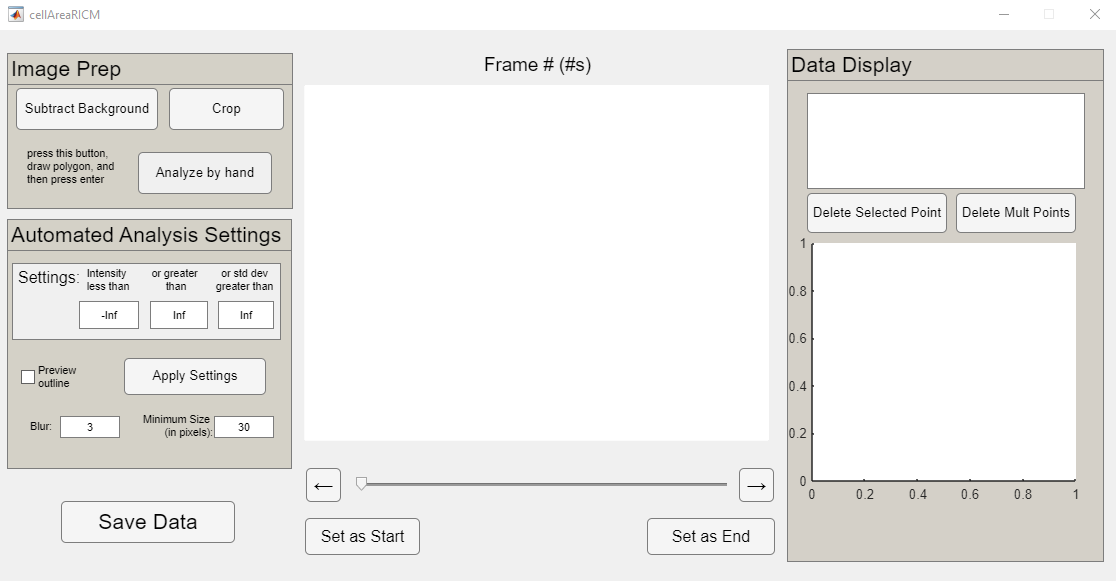
**Instructions for use of cellAreaRICM**:

cellAreaRICM is a MATLAB app designed for analysis of single spreading cells from reflection interference contrast microscopy (RICM) images of the cell-substrate contact region over time. The GUI was originally written using GUIDE in an earlier version of MATLAB, but was most recently converted to a MATLAB app in MATLAB 2022a.

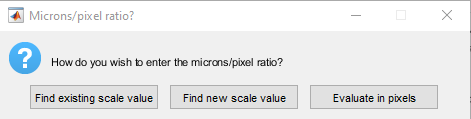
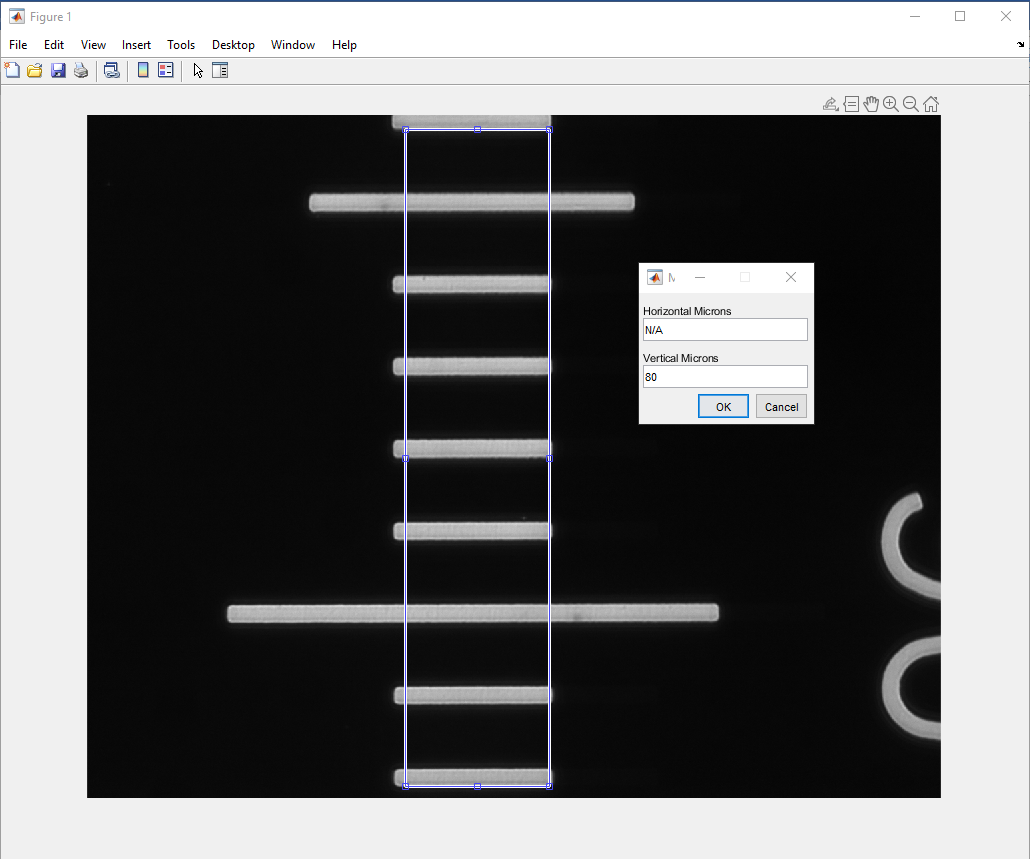
The program is currently only written to analyze a single contact region at a time, but could be configured in the future to analyze multiple spreading cells in parallel.

The application window appears like this:

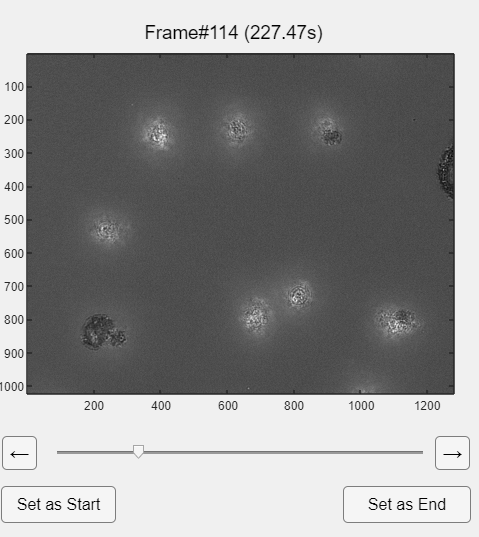


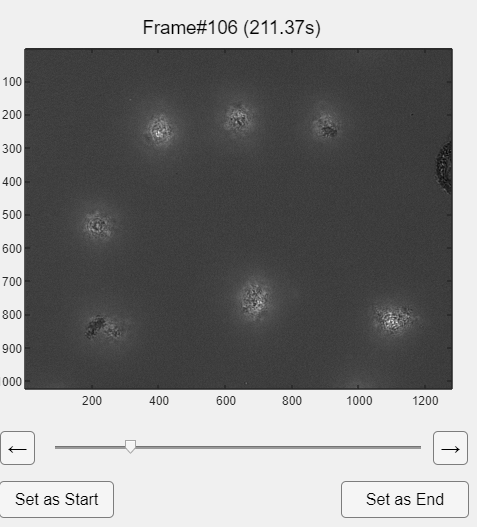
In general, the workflow for analyzing a spreading contact region is as follows:

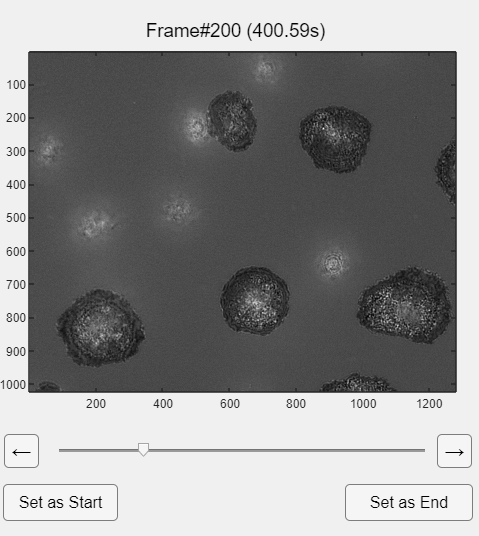
1. Upon opening the application, a dialog window will open prompting you to choose the first image of the RICM image sequence. This can be either a .bmp or a .tiff image. All images in the same folder as the selected image will be used for the complete sequence. If a dialog box also requests a “time file” and you do not have a time txt file saved from programs in the Heinrich lab, exit from that dialog and select “Use frame number” in the next popup. All time values will actually be frame numbers in that case.
2. An additional dialog box then opens to set the scale value:

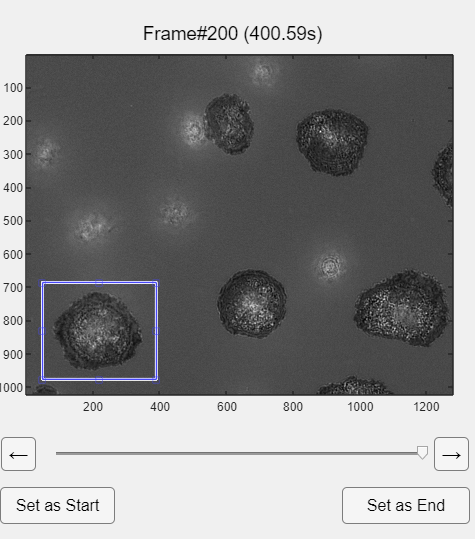
  
If a scale value has been saved from a previous analysis, this can be loaded using the first option. To measure a new scale value, select the middle option and then a dialog box will open, allowing you to load a scale image. You should then draw a rectangle on the scale bar image with a known width and/or height, as shown:  
   
After drawing this box, press enter and a dialog box will open asking you to enter horizontal and vertical micron values. In the example shown, only the vertical value is known, so it is entered and the “horizontal microns” value is left as the default “N/A”. The next dialog box allows you to save the scale value as a .mat file for future use.

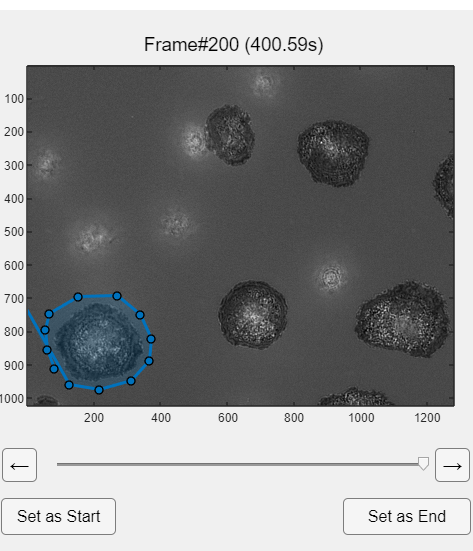
1. *Optional, but recommended*: If possible, complete background subtraction. This can be done by pressing the top left button and opening your best background image when prompted. The background is filtered using a low-pass Wiener filter to remove noise, and then is subtracted. Note that the average pixel value after subtraction is set to 0 (so some values are negative), whereas if background subtraction is not completed, all pixel values are positive.
2. You can now navigate through the image sequence using the slider and arrow buttons:

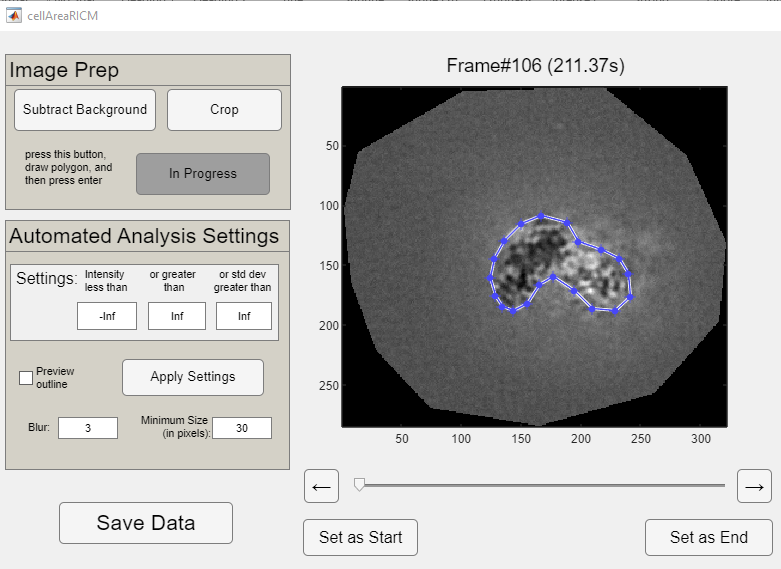
  
The plot title is set to display frame number and time (will be identical to the frame number if you do not have a time .txt file from the Heinrich Lab). Navigate through the sequence to identify a cell of interest. In this example, I will use the one in the bottom left of the sample image above.

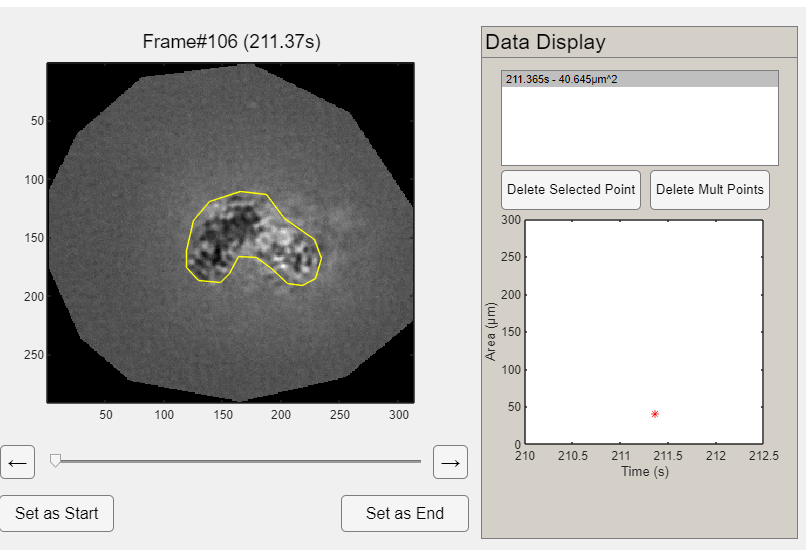
1. Upon identifying the cell of interest, find the first and last usable frames. In this case, I navigate to the first frame where this cell comes into view, and then select “set as start”:  
     
   And then I go to the final frame (after which another cell interferes with the view of this one) and press “set as end”:



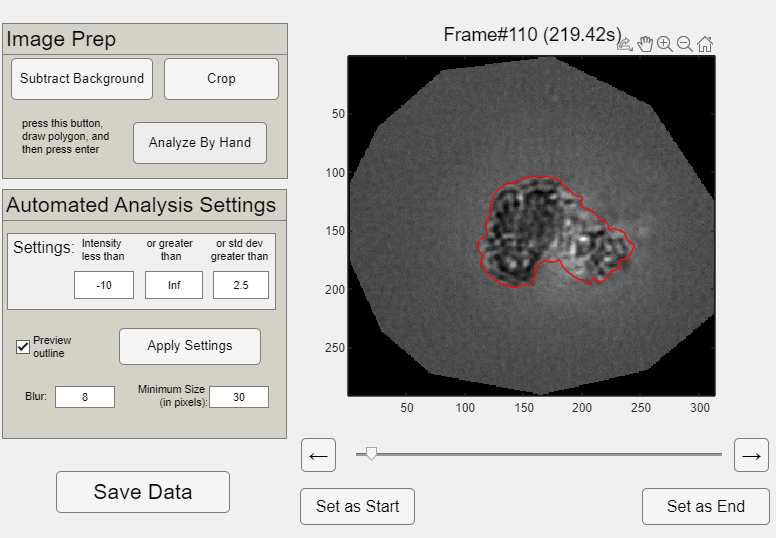
1. I then press “crop” in the top left panel to isolate the single cell of interest. Generally, this should be done using the final frame of analysis, in which the cell has spread its furthest. The crop can be done with a typical rectangle, then double clicking in the selected region  
     
   Or by selecting a polygonal crop region (in the code, all pixels outside the polygon are ignored by setting them to NaN):

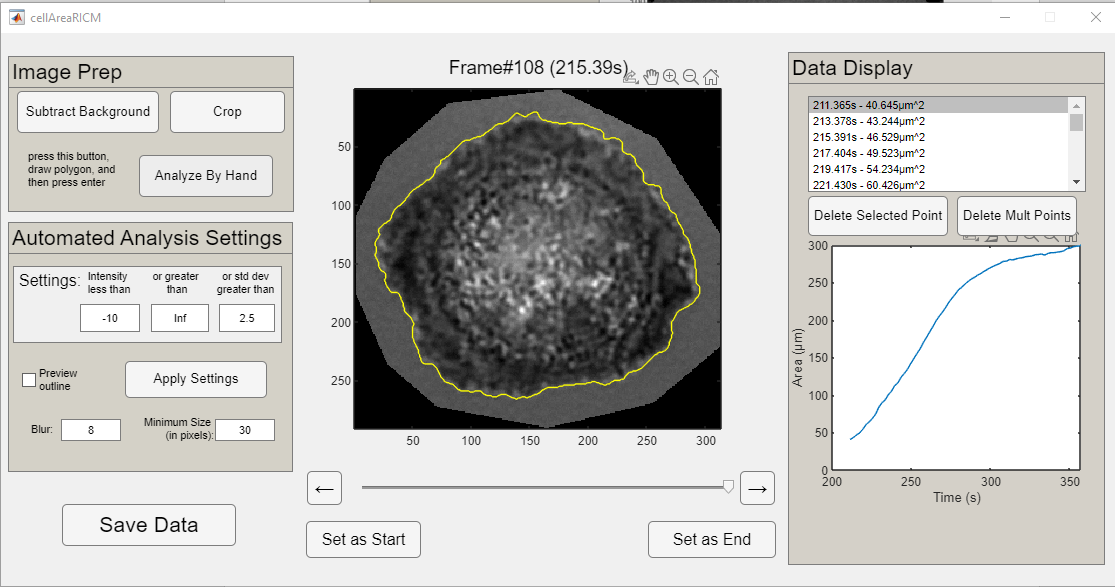
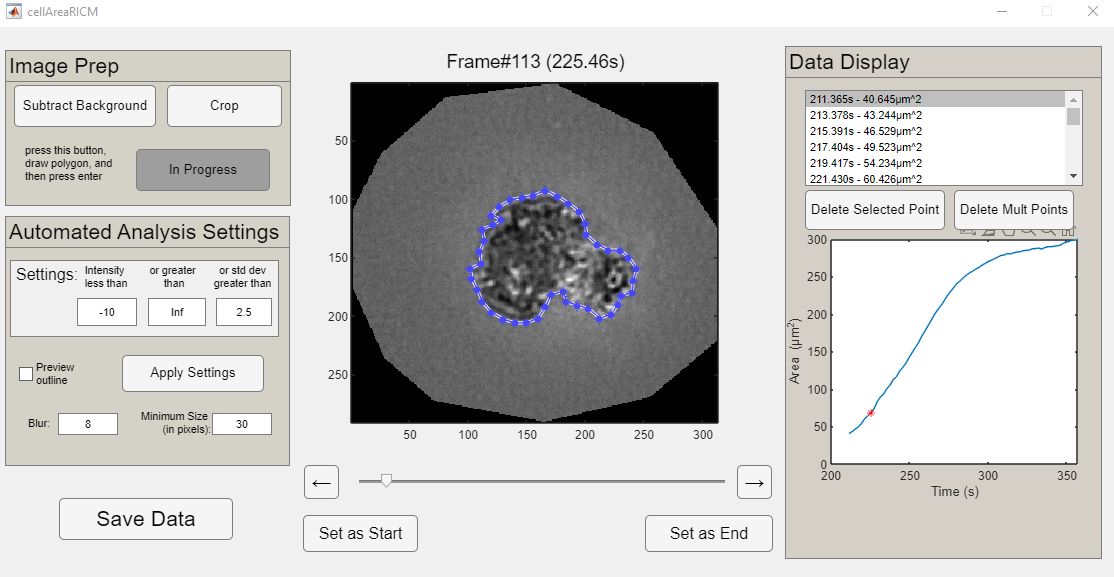


1. In most cases, the initial images of contact area growth need to be traced by hand, as our simple thresholding algorithm does not perform reliably in these cases. To do this, simply press the “Analyze by hand” button, trace a polygon around the contact region, adjust if necessary, and then press enter (note the button reads “in progress” until you press enter):  
     
   A dialog box will then open asking you to start saving a text file with contact area values. The location to save this should ideally be its own folder, as several files will be saved here (see notes about “save data” at end).  
     
   The data display will now show the analyzed contact area:

  
The red asterisk point on the graph indicates the current data point, and the exact measured contact area is listed in the list box on the top right. Points can be selected and deleted if desired.

1. The automated analysis parameters can be tuned to the current cell by selecting “preview outline” and adjusting the values in the automated analysis box:

  
As stated in the box, the algorithm thresholds on both intensity and local standard deviation (I generally only use the “intensity less than” and “std dev greater than” options as shown). The user can also specify a minimum size for detection (would be more applicable in a future iteration for which multiple cells are analyzed) and the blur factor. The blur factor corresponds to the radius of the Gaussian kernel, which chosen to be 3 times the standard deviation of the Gaussian kernel here. Higher blur results in smoother, but less detailed contours.

1. Once all the “by hand” analysis has been completed, and the automated analysis settings have been refined, press “apply settings”. The last frame analyzed by hand dictates where the automated analysis starts (at the next frame after that).  
     
   If you want every frame after the last one analyzed by hand to be analyzed, enter “1” in the next dialog box, if you want every 1 frame out of every 3 to be analyzed, enter “3”, etc…  
     
   The analysis then begins with progress shown in a loading bar. Once completed, the full plot is shown:  
   
2. *If necessary*: The analysis can be further refined by deleting errant points using the buttons on the right panel, or by fixing errant points by navigating to the point of concern, and then pressing “analyze by hand”. This brings up the current shape, but allows you to move or delete (right click to see this option) given vertices:  
     
   Press enter after finishing this “by hand” alteration.
3. Finally, the full data can be saved using the bottom left “Save Data” button. An input dialog box will allow you to enter additional notes to be saved in the log file for this cell. You then choose a location to save the log file. The complete list of files saved is:
   1. Contact-area-versus-time file: Text file with two columns – time and contact area. Saved throughout the analysis (location set in step 7)
   2. *If applicable*: background image used for this analysis, saved in same folder as CA vs. time
   3. Cell location file: Image showing which region of the source image was cropped, always named “cellLocationRICM.bmp”
   4. Analysis info file: Txt file named “analysisInfo” followed by the file name of the CA vs time file. Gives the source and saving directories, file type, scale value, crop coordinates, polygon coordinates from the hand-drawn analysis, a list of the frames analyzed using the automated settings, and the automated analysis settings themselves.
   5. Log file: only saves extra notes currently.
   6. “Cell prints” file: a MATLAB 3D binary matrix saved as a .mat file, named “cellPrints” followed by the name of the CA vs time file. The matrix is *i* by *j* by *k*, where *i* by *j* is the cropped image dimensions and *k* is the total number of frames. “True” indicates pixels belonging to the contact region and “false” indicates other pixels.

Any questions on the program and any bugs should be reported to Emmet Francis at emmet.a.francis@gmail.com.