Analyzing bowels using this code: A step-by-step guide

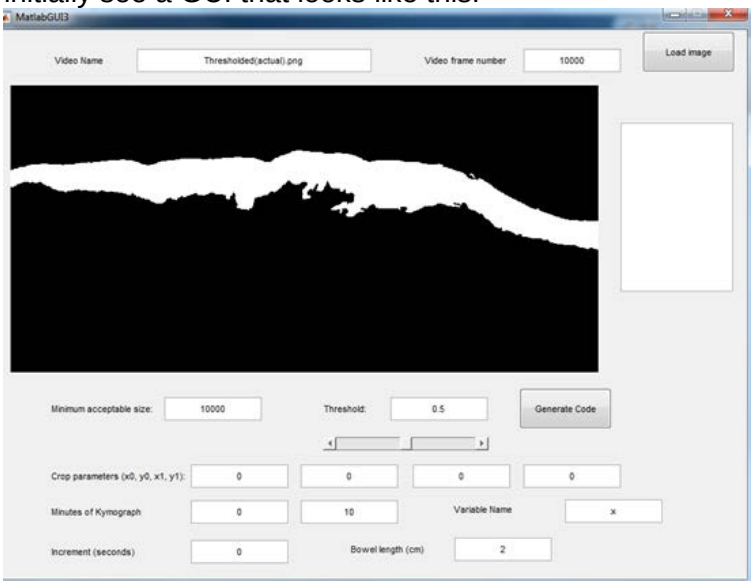
Collecting and processing video

1. First, film the bowel in the orientation shown below. Make sure the background is much darker than the bowel (a black sheet beneath the bowel is recommended). An example screenshot from a good bowel video is included below for reference. Some bubbles are okay:

2. Make sure you know the scale of your bowel. It is usually idea to take at least one picture/movie with a ruler.
3. Ensure that the video is saved as a video file that Matlab can read, e.g. an .mp3 file or a .wmv
4. If it is not saved as this file type, programs are available online that can convert the videos to the appropriate type.
5. I recommend saving an additional copy of the video at 16x, for validation purposes (it is much easier to watch a 20 minute video for colon contractions at 16x than at 1x).

Using ThresholdAndCropGUI to define parameters for cropping the bowel

1. Before you can generate the kymograph, you need to define some optimal parameters for thresholding your movie.
2. To do this, you will need to run ThresholdAndCropGUI.m
3. You should initially see a GUI that looks like this:



4. First, input your video file name and a video frame number. Hit "Load image." (If you do not do this, a default image appears that you can play with).
5. Adjust minimum acceptable size until you achieve an acceptable degree of bowel segmentation. Minimum acceptable size defines the minimum area of white, thresholded objects within your picture. It helps get rid of bubbles or other extraneous objects that are not touching your bowel.

   Bowel with minimum acceptable area 100:

Bowel with minimum acceptable area 100000:

6. Adjust the threshold until you get a threshold that accurately reflects your bowel's shape. Each pixel in the image is scaled from 0 to 1, with 0 being black and 1 being white. Thresholding to 0.5 accepts every pixel above intensity 0.5. Therefore, lower thresholds will be more permissive, whereas higher thresholds will be less permissive.
7. If there is extraneous stuff in your image, adjust the crop parameters until you have cut out the unnecessary stuff.
8. Define when you want to start and end the kymograph by inputting values into "Minutes of Kymograph." For instance, if you want to start the kymograph at minute 1 and end at minute 9, you would input 1 and 9. Make sure there are enough minutes in your video! Do not try to start at minute 1 and end at minute 10 if there are only 8 minutes in your video.

9. Define what increment you want to sample to make your kymograph. If you want to sample once every second, input 1 into this field. If you want to sample once every half second, input 0.5.
10. Input a unique variable name for each video. This will be a handle for generating variables containing kymograph data and other information. Variable names cannot start with numbers.
11. Finally, input the length in centimeters that corresponds to the width of your video. For instance, if your video's width spans 2 centimeters, input 2 into this field.
12. Click the Generate Code button, and save the file.
13. A table with more information on the different fields and their significance is below:

Field	**What it means**
Video name	The name of the video file. For example: 00170 – 9055 – last 20 min.wmv
Video frame number	A representative frame of your video that you want to try to threshold. Usually defaulting to Frame 1 is fine, unless the first couple seconds of your video are not representative of the rest of your video.
Minimum acceptable size	Minimum acceptable size defines the minimum area of white, thresholded objects within your picture. It helps get rid of bubbles or other extraneous objects that are not touching your bowel.
Threshold	Each pixel in the image is scaled from 0 to 1, with 0 being black and 1 being white. Thresholding to 0.5 accepts every pixel above intensity 0.5. Therefore, lower thresholds will be more permissive, whereas higher thresholds will be less permissive.
Crop parameters (x0, y0, x1, y1)	These parameters can be adjusted in order to remove unnecessary objects or parts of your image. However, make sure if you adjust the width of the image that you modify the bowel length parameter accordingly.
Minutes of Kymograph	When you want to start and end your kymograph. If you want to start the kymograph at minute 1 and end at minute 9, you would input 1 and 9, for 8 total minutes of video. Make sure there are enough minutes in your video! Do not try to start at minute 1 and end at minute 10 if there are only 8 minutes in your video. (If you do this, it is not the end of the world, but the y axis/time values for your kymograph might be messed up.
Increment (seconds)	Defines the sampling increment for your kymograph. If you want to sample once every second, input 1 into this field. If you want to sample once every half second, input 0.5.
Variable name	This will be a handle for generating variables containing kymograph data and other important information. Make a unique variable name for each video. Variable names cannot start with numbers.
Bowel length (cm)	The length in centimeters that corresponds to the width of your video. For instance, if your video's width spans 2 centimeters, input 2 into this field.

Generating kymograph matrices:

1. Open the file that you just saved. It contains code that can be run on the command line to generate the matrix of data to make your kymograph.

2. Because it can take a couple hours to generate the kymograph matrices (I promise, I am working on making this faster…) I recommend combining code for your entire set of bowels into a single file and running it all on the command line overnight. For example, I generated kymographs for the following 3 bowels all at once by pasting the following onto the command line:

```
TsDn4365firstvidR = VideoReader('00091 - 4365 - first 20 min.wmv');
[TsDn4365firstmatrx, TsDn4365firstmax, TsDn4365firstmin,
TsDn4365firstx, TsDn4365firsty] = generatekymograph(TsDn4365firstvidR,
[0   100  1920   950], 0, 20, 1, 0.2, 2.2, 100000);

TsDn4365nextvidR = VideoReader('00092 - 4365 - next 20 min.wmv');
[TsDn4365nextmatrx, TsDn4365nextmax, TsDn4365nextmin, TsDn4365nextx,
TsDn4365nexty] = generatekymograph(TsDn4365nextvidR,
[0   100  1920   950], 0, 20, 1, 0.2, 2.2, 100000);

TsDn4365lastvidR = VideoReader('00093 - 4365 - last 20 min.wmv');
[TsDn4365lastmatrx, TsDn4365lastmax, TsDn4365lastmin, TsDn4365lastx,
TsDn4365lasty] = generatekymograph(TsDn4365lastvidR,
[0   100  1920   950], 0, 20, 1, 0.2, 2.2, 100000);
```

3. After running this code, the following will be saved in your Matlab Workspace for each video:
    a. [Variable name]vidR: A VideoReader object.
    b. [Variable name]matrx: The kymograph data, a 2-dimensional matrix containing bowel widths as a function of time and distance down the bowel.
    c. [Variable name]max: The maximum value in your kymograph
    d. [Variable name]min: The minimum value in your kymograph
    e. [Variable name]x: A 1-dimensional array containing x axis number labels.
    f. [Variable name]y: A 1-dimensional array containing y axis number labels.

<u>Displaying the kymographs:</u>

1. SAVE YOUR WORKSPACE.
2. Double check that the appropriate values were saved in your Matlab Workspace and that no errors occurred overnight.
3. Right now, the kymograph data matrix is in units of pixels (number of pixels of bowel width at each time point). However, we want this information in millimeters. So we are going to convert it. The conversion factor is total bowel width in the video * 10/pixels of video. To find this last value, type [Variable name]vidR into the command line. You should see a "Width" parameter listed under "Video properties."
4. Modify the following code as appropriate and run it on the command line to display the kymograph of interest:

```
[Variable name]matrxDistInmm = [Variable name]matrx*([Conversion factor that
you just calculated]);
fig=figure;
colormap(jet);
imagesc([Variable name]x, [Variable name]y, [Variable name]matrxDistInmm);
xlabel('Distance down bowel (cm)');
ylabel('Time (m)');
h = colorbar();
ylabel(h, 'Bowel width (mm)');
caxis([1.5 4.5]);
saveas(fig,'Figures\[Variable name].png');
saveas(fig,'Figures\[Variable name].fig');
```

5. Feel free to play around with the color axis (caxis) range (in mm) until you find a range that makes sense.
6. Congratulations! You are done.