

```
1 % this is the segmentation portion of my code.
2
3 clear all; % clears running programs for this program only
4 close all;
5 %% these are for iterative processing to show each stage when code is ran
6
7 Img=imread('raw00.tif'); % image to be segmented
8 Img2=imread('mitochondria00.png'); % ground truth to compare output
9
10 subplot(1,2,1), imshow(Img); %sets first figure to show original
11 image
12 subplot(1,2,2), imshow(Img2); % this is place holder for each
13 segmented image
14 %% creates a textured image and rescales it with mat2gray()
15 E = entropyfilt(Img);
16
17 Eim = mat2gray(E);
18 imshow(Eim);
19
20 %% thresholds the rescaled image texture
21 BW1 = im2bw(Eim, .8); % the .8 here is the "magic number" to
determine the texture to be segmented
22
23 imshow(BW1);
24 figure, imshow(Img);
25
26 %% extracts the texture
27 BWao = bwareaopen(BW1,2000);
28 imshow(BWao);
29
30 %% smooths it and closes open holes
31 nhood = true(9);
32 closeBWao = imclose(BWao,nhood);
33 imshow(closeBWao)
34
35 %% fills in more holes and makes the roughMask for the texture
36 roughMask = imfill(closeBWao,'holes');
37
38 imshow(roughMask);
39 figure, imshow(Img);
40
41 %% gets the raw of remaining textures
42 I2 = Img;
43 I2(roughMask) = 0;
44 imshow(I2);
45
46 %% creates a texture image and rescales it as before
47 E2 = entropyfilt(I2);
48 E2im = mat2gray(E2);
49 imshow(E2im);
50
51 %% thresholds "2nd" textures
52 BW2 = im2bw(E2im,graythresh(E2im)); % uses graythresh as opposed to
.8
53
54 imshow(BW2)
55 figure, imshow(Img);
56
57 %% gets second mask
58 mask2 = bwareaopen(BW2,1000);
59 imshow(mask2);
60
61 %% using the two masks created from previous code it extracts them
from Img
62 texture1 = Img;
```

```
63     texture1(~mask2) = 0;
64     texture2 = Img;
65     texture2(mask2) = 0;
66
67     imshow(texture1);
68     figure, imshow(texture2);
69
70
71 %% Places a boundary between segmented texture and rest of image
72 boundary = bwperim(mask2);
73 segmentResults = Img;
74 segmentResults(boundary) = 255;
75 imshow(segmentResults);
76
77 %
78 %% Alternative to entropyfilt()
79 S = stdfilt(Img,nhood);
80 imshow(mat2gray(S));
81
82 R = rangefilt(Img,ones(5));
83 imshow(R);
84 %
85 %
86 %
87 %% this code is for some testing of functions
88 Img=imread('csCircles.jpg');
89
90 %imshow(Img);
91
92     Img = im2double(Img);
93     [r c p] = size(Img);
94
95     imR = squeeze(Img(:,:,1));
96     imG = squeeze(Img(:,:,2));
97     imB = squeeze(Img(:,:,3));
98
99     imBinaryR = im2bw(imR, graythresh(imR));
100    imBinaryG = im2bw(imG, graythresh(imG));
101    imBinaryB = im2bw(imB, graythresh(imB));
102    imBinary = imcomplement(imBinaryR&imBinaryG&imBinaryB);
103    imshow(imBinary);
104
105    [labels, numLabels] = bwlabel(imBinary);
106    disp(['Number of objects detected: ' num2str(numLabels)]);
107 %
```