Table of Contents

Find Fat Droplets in Histology Slide

Susan Noworolski 1/21/2018 Use this code as a starter. Places you need to modify the code are marked with the following (and may have some lines started, but commented out):

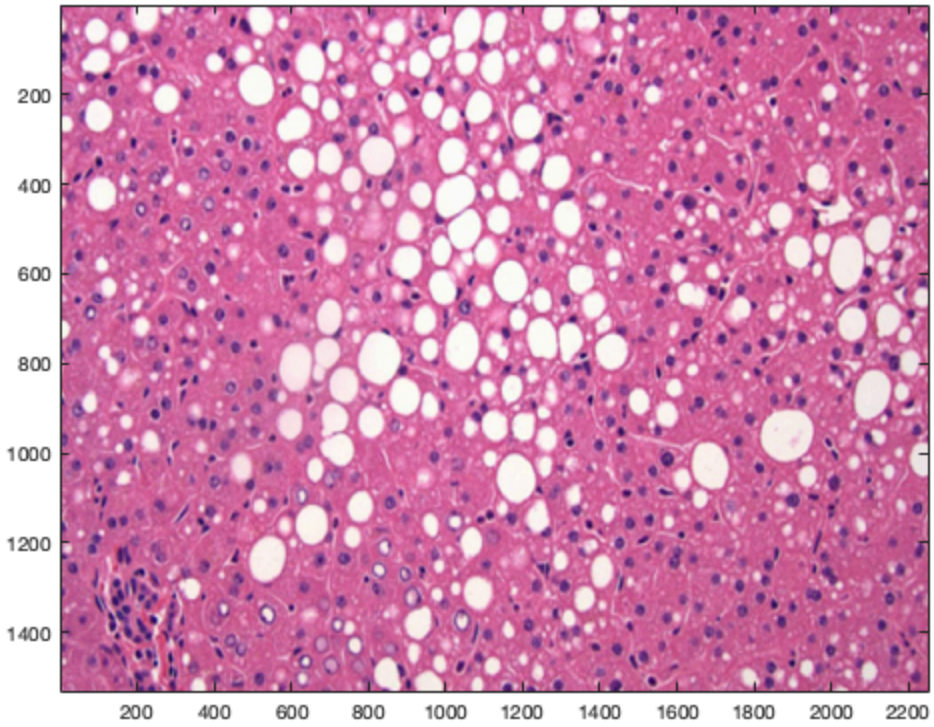
```
%>>>>>>>>>>
```

```
%<<<<<<<<<<<<
```

Initialize

```
clear all; close all;
Im=imread('histology_droplets.jpg');
%Im=imresize(Imin,4);
figure('name','Histology Slide');
imagesc(Im);
%Set min,max radii and sensitivities for imfindcircles:
%>>>>>>>>>
minr = 28;  %minimum big droplet radius to identify
maxr = 100;  %maximum big droplet radius to identify
sens = 0.8; %sensitivity for identifying big droplets
```

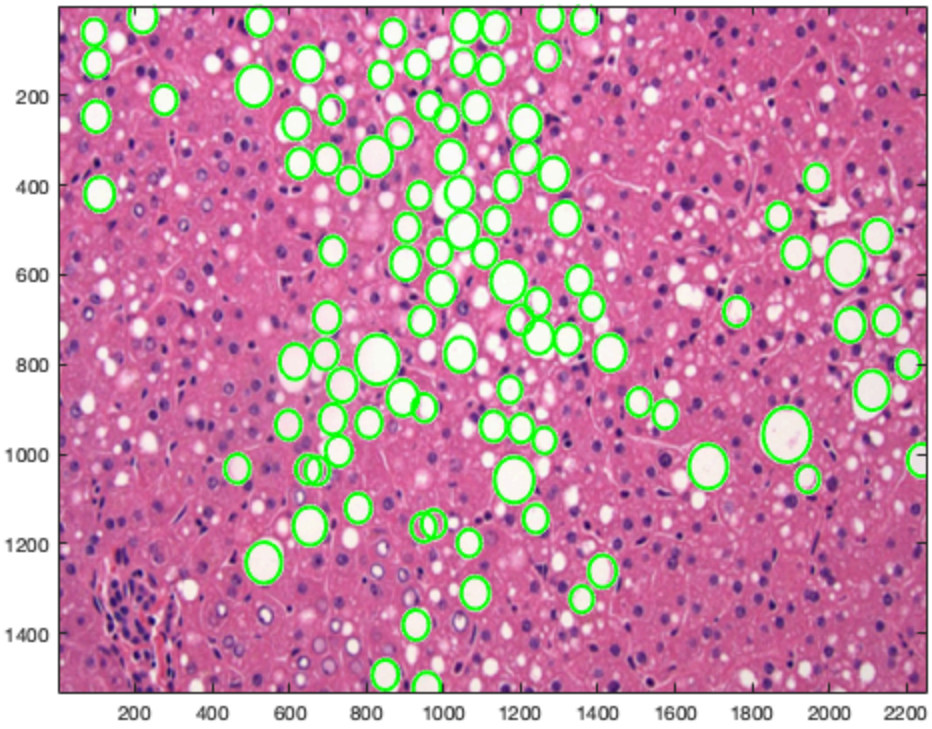
```
%<<<<<<<<<
```



Find Big Droplet Fat

```
close all;
figure('name','Histology Slide - Large Droplet Fat');
imagesc(Im);
[centersbig,radiibig,metricbig] = imfindcircles(Im,
[minr,maxr], 'Sensitivity',sens);

% - Display big droplets on array
viscircles(centersbig, radiibig, 'Edgecolor', 'g');
```

```
Warning: You just called IMFINDCIRCLES with a large radius range.
 Large radius
ranges reduce algorithm accuracy and increase computational time. For
 high
accuracy, relatively small radius range should be used. A good rule of
 thumb is
to choose the radius range such that Rmax < 3*Rmin and (Rmax - Rmin) <
 100. If
you have a large radius range, say [20 100], consider breaking it up
 into
multiple sets and call IMFINDCIRCLES for each set separately, like
 this:

 [CENTERS1, RADII1, METRIC1] = IMFINDCIRCLES(A, [20 60]);
 [CENTERS2, RADII2, METRIC2] = IMFINDCIRCLES(A, [61 100]);
```

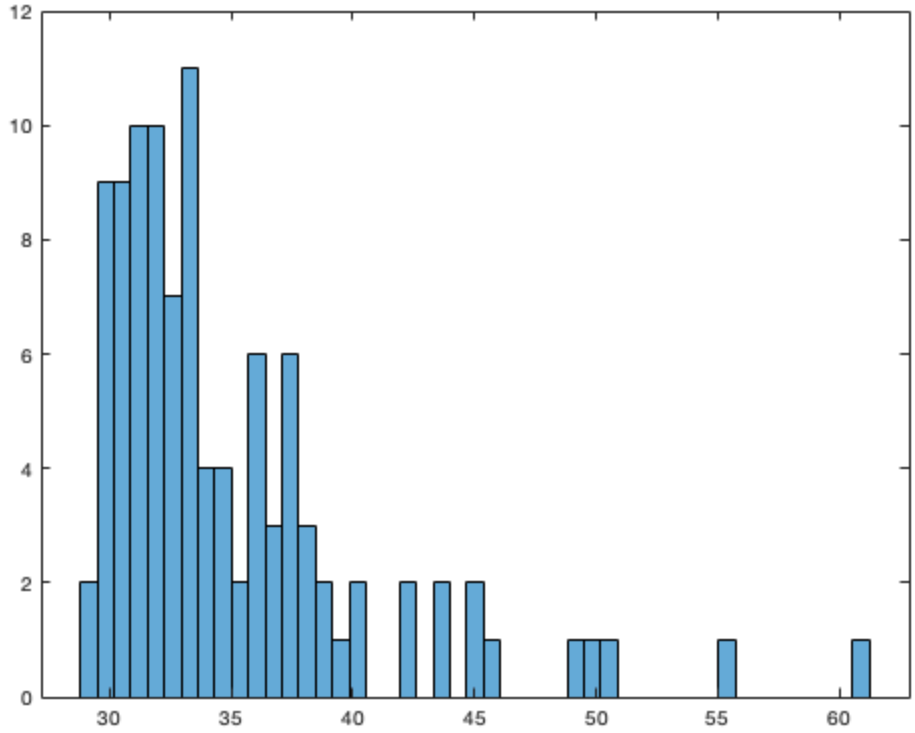


Calculate # big circles, mean radius, median radius, histogram big droplet radii

```
%>>>>>>>>>>>
```

```
numbigcir=size(centersbig,1);
mean_r= mean(radiibig);
median_r = median(radiibig);
bigdrop_r_hist = histogram(radiibig,47);
```

```
%<<<<<<<<<<
```



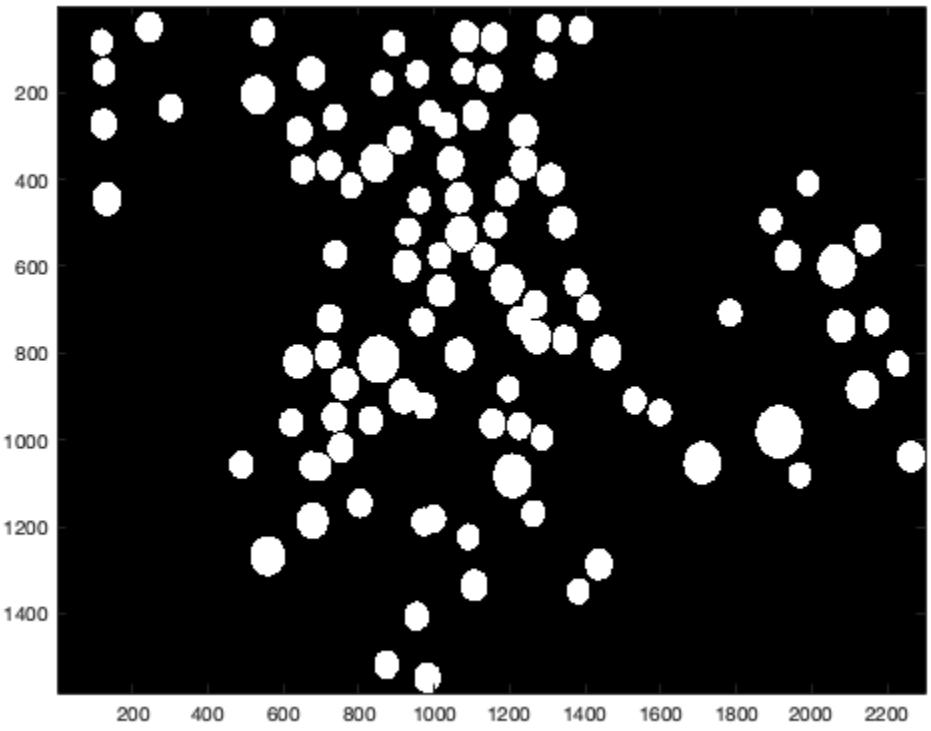
Set up blank array to paint segmented circles to calculate areas

```
close all;

% Some circles are not fully within Im array (fall off edges) -->
 increase size of array to
% include complete circles
%
%>>>>>>>>>>
offset=50;
k=round(offset/2);

%Set up arrays to paint droplets
blank = zeros(size(Im,1)+offset, size(Im,2)+offset);
```

Paint Big Droplets on Array

```
%Loop through all circles, painting circles as '1', background has '0'
bigdrop = blank; % start with a blank slate
for i=1:size(centersbig,1)
    th = linspace(0,2*pi,3000); % Step through enough angles to get
 ~full circumference
    r=radiibig(i);
```

```
        %Fill manually, issue w/ round off errors; may not get all pixels
        %Burn pixels as concentric circles, from center to radius
        for j=1:round(r)*2
            x = j/2.0*cos(th) +centersbig(i,1);
            y = j/2.0*sin(th) +centersbig(i,2);
            for w = 1:length(y)
                row=round(x(w))+k;
                col=round(y(w))+k;
                bigdrop(col,row)=1;
            end
        end
    end

    %--------------------------------
    figure('name','Blank image with filled circles');
    imagesc(bigdrop);
    colormap gray;
```

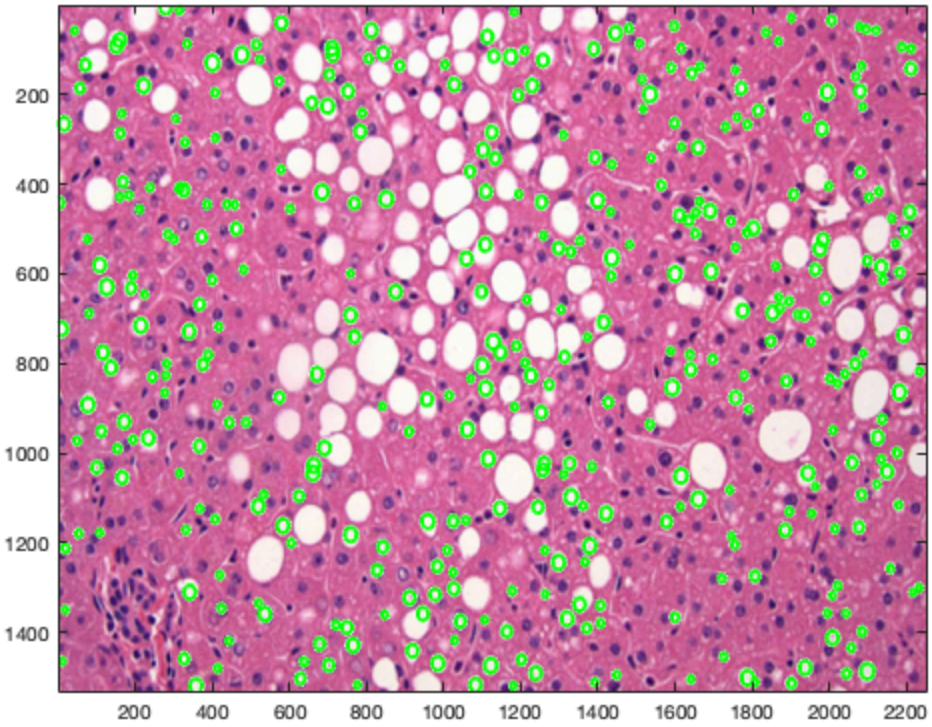


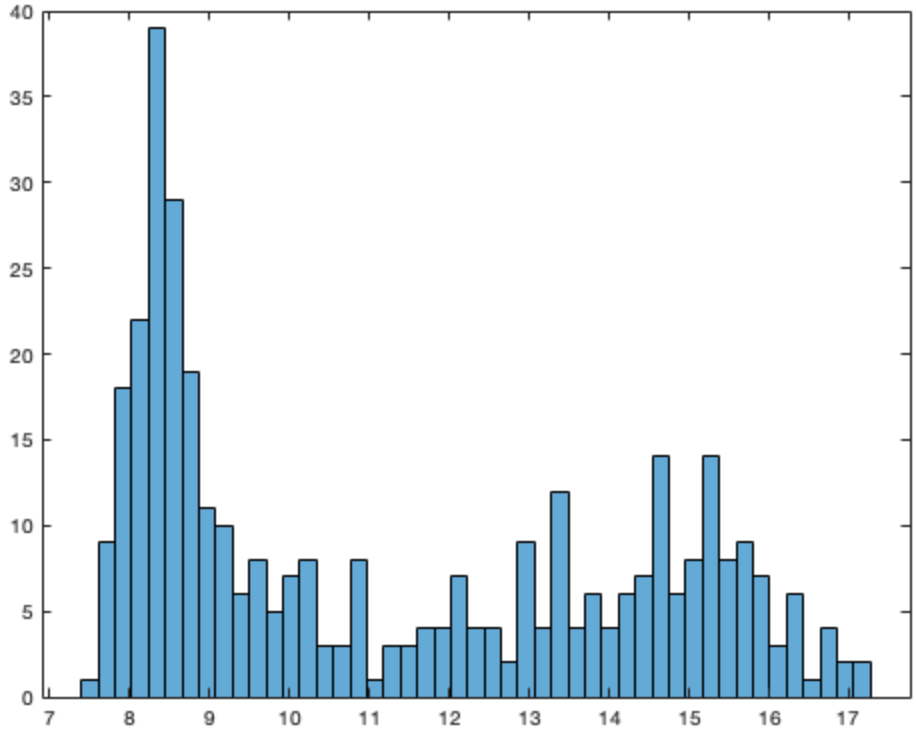
Find Small Droplet Fat

Repeat similar to the Find Large Droplet Fat section; Use different radii, possibly sensitivity (Can copy code and modify with new variable names or make a function/s)

```
%close all;
%>>>>>>>>>>>>>
%Find Small Droplet Fat
```

```
minr = 7;  %minimum small droplet radius to identify
maxr = 18;  %maximum small droplet radius to identify
sens = 0.775; %sensitivity for identifying small droplets
figure('name','Histology Slide - Small Droplet Fat');
imagesc(Im);
[centerssmall,radiismall,metricsmall] = imfindcircles(Im,
[minr,maxr], 'Sensitivity',sens);

%Display circles
viscircles(centerssmall, radiismall, 'Edgecolor', 'g');
% Calculate # small circles, mean radius, histogram small droplet
 radii
numsmcir=size(centerssmall,1);
mean_r_sm= mean(radiismall);
median_r_sm = median(radiismall);
figure()
smdrop_r_hist = histogram(radiismall,47);
%<<<<<<<<<<<<<<
```

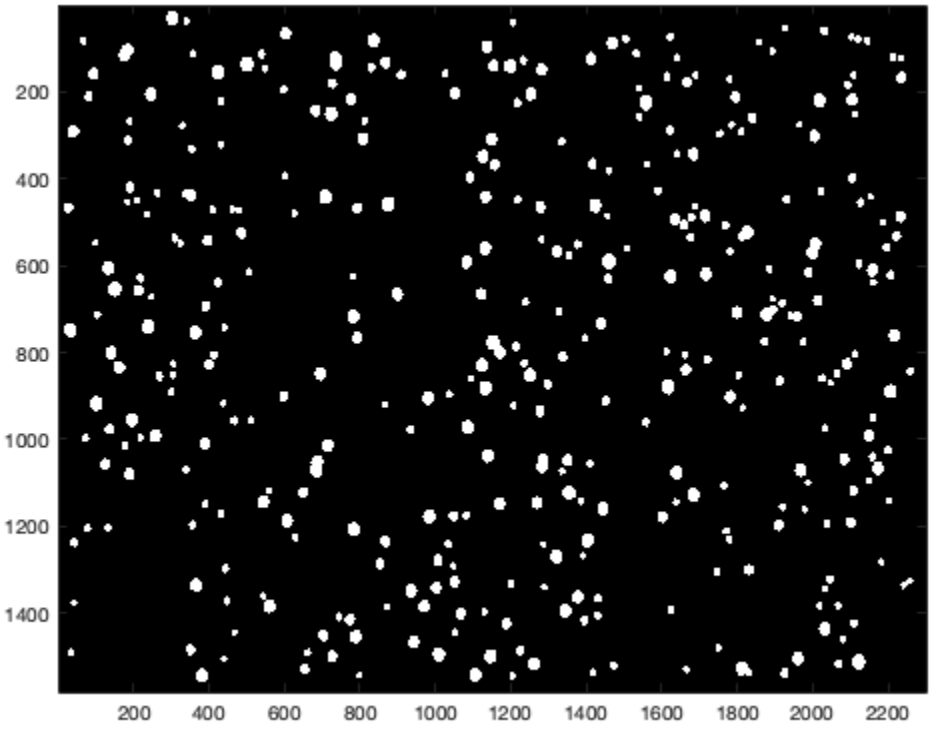




Paint Small Droplets on Array

```
%Repeat similar to Paint Large Droplets on  a blank array (Can copy
 code and modify with new variable
% names or make a function/s)
%>>>>>>>>>>>>>>>>>>>
%Loop through all circles, painting circles
smalldrop = blank; % start with a blank slate
for i=1:size(centerssmall,1)
    th = linspace(0,2*pi,3000); % Step through enough angles to get
 ~full circumference
    r=radiismall(i);
    %Fill manually, issue w/ round off errors; may not get all pixels
    %Burn pixels as concentric circles, from center to radius
    for j=1:round(r)*2
        x = j/2.0*cos(th) +centerssmall(i,1);
        y = j/2.0*sin(th) +centerssmall(i,2);
        for w = 1:length(y)
            row=round(x(w))+k;
            col=round(y(w))+k;
            smalldrop(col,row)=1;
        end
    end
end
%Display blank array with filled small droplet circles
```

```
%--------------------------------
figure('name','Blank image with filled circles');
imagesc(smalldrop);
colormap gray;


%<<<<<<<<<<<<<<<<<
```



Calcuate Stats on segmented circles / droplets

```
%You may have different variable names than listed for the small
 droplet
%fat variables. Update as needed.

%Zoom to cut off border
cropx = size(smalldrop,1)-k;
cropy = size(smalldrop,2)-k;
%Make mask arrays the same size as the original image
%(circles = '1', background = '0')
cropsmdrop = smalldrop(k:cropx, k:cropy);
cropbigdrop = bigdrop(k:cropx, k:cropy);

%Calculate area of small & large droplets in pixels and in % of the
 slide
%pixels
%>>>>>>>>>>
```

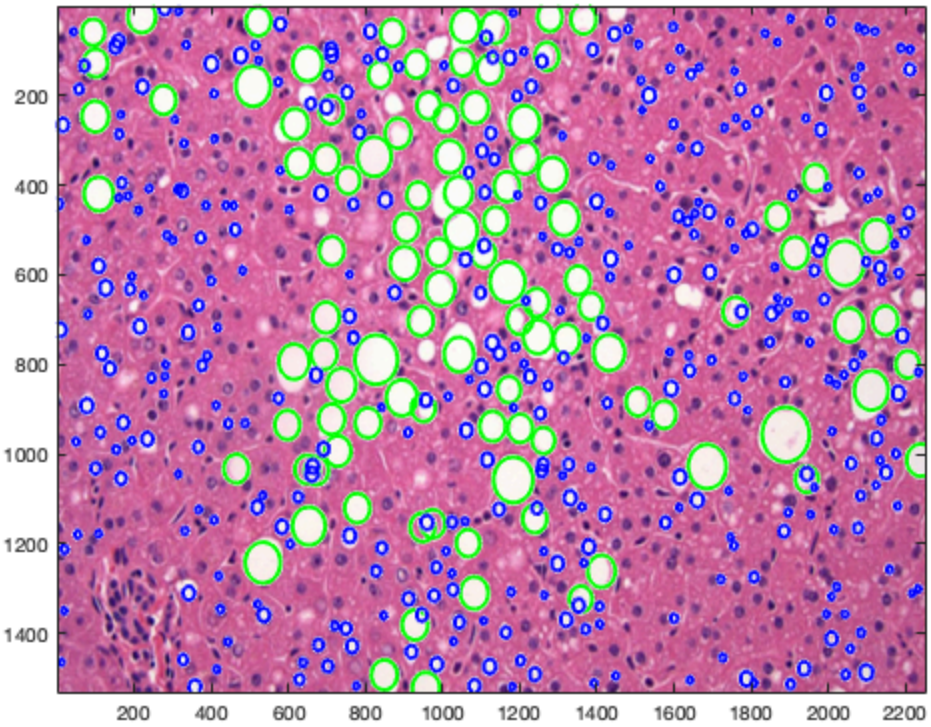
```
big_area_pix= 0;
for i = 1:size(bigdrop,1)
    for j = 1:size(bigdrop,2)
        if bigdrop(i,j) == 1
           big_area_pix=big_area_pix+1;
        end
    end
end

big_area_percent = big_area_pix/(size(Im,1)*size(Im,2));

small_area_pix= 0;
for i = 1:size(smalldrop,1)
    for j = 1:size(smalldrop,2)
        if smalldrop(i,j) == 1
           small_area_pix=small_area_pix+1;
        end
    end
end

small_area_percent = small_area_pix/(size(Im,1)*size(Im,2));

%Calculate totals: number, area, average radii, median radii, and
 total fat %area (of slide
%area):
total_droplet_pix = big_area_pix+small_area_pix;
total_droplet_area = big_area_percent + small_area_percent;
all_radii = vertcat(radiibig,radiismall);
avg_radii = mean(all_radii);
med_radii = median(all_radii);

%<<<<<<<<<<<<<
```

Summary of Large and Small Droplet Fat Segmentation

```
%Display both large and small droplets on the histology slide
figure('name','Histology Slide - Large (green) and Small (blue)
 Droplet Fat');
imagesc(Im);
%>>>>>>>>>>>>>>>
%Display circles
viscircles(centersbig, radiibig, 'Edgecolor', 'g');
viscircles(centerssmall, radiismall, 'Edgecolor', 'b');

%<<<<<<<<<<<<<<<<

%Display a summary of results
disp(sprintf('Size  Number Total_Area Avg_Radius Median_Radius %
%Slide'))
disp(sprintf('Large 103   417405 pixels 35.0148   33.2767   12.13
 percent '))
```

```
%Repeat for small droplets/circles and for total circles
%>>>>>>>>>>>>>>>

disp(sprintf('Size  Number Total_Area Avg_Radius Median_Radius %
%Slide'))
disp(sprintf('Small 374   174066 pixels 11.2610   10.2185   5.06
 percent '))
%<<<<<<<<<<<<<<<
```

```
Size  Number Total_Area Avg_Radius Median_Radius %Slide
Large 103   417405 pixels 35.0148   33.2767   12.13 percent
Size  Number Total_Area Avg_Radius Median_Radius %Slide
Small 374   174066 pixels 11.2610   10.2185   5.06 percent
```



Questions

```
%Answer the questions here or separately - whichever is easier.
% 1. Well, some small droplets are within the larger ones. But at the
 same
% time, some large droplets aren't fully captured. Overall, it should
 be a
% resonable estimate unless you have too much overlap.
% 2. In the large droplet, larger droplets that aren't circular would
 not
```

```
% have their entire shape captured. Smaller large droplets with
 irregular shapes
% would have too much area captured. For smaller droplets, some would
% overlap with larger droplets and be repeated.
% 3. With smaller droplets, I used a lower sensitivity. With a higher
% sensitivity, too much was captured. With larger droplets if the
% sensitivity was too low not enough of the area would be captured.
```

Published with MATLAB® R2018b