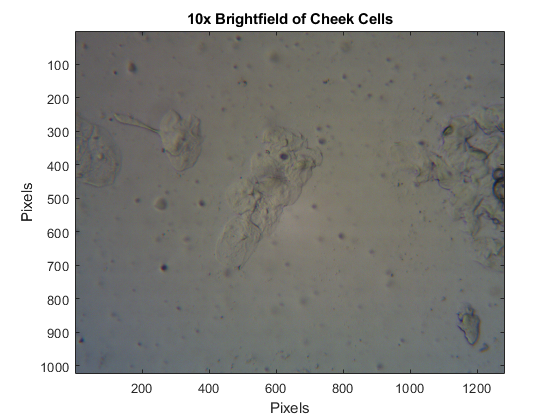
Vibhav Jha

BE 492 Section B5a

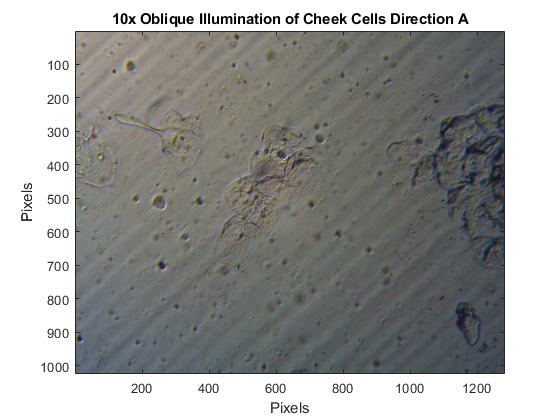
April 1, 2019

Lab 4: Imaging Structure and Dynamics

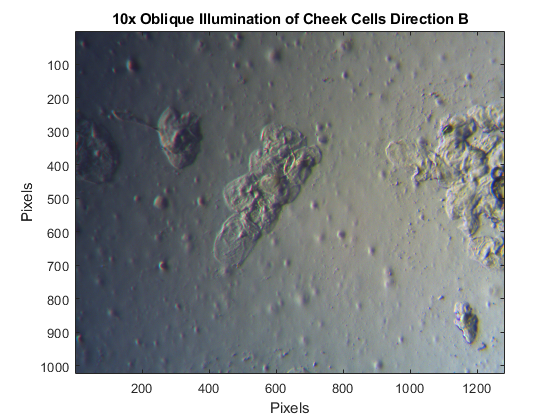
**Part B: Contrast Mechanisms based on scattering**

****

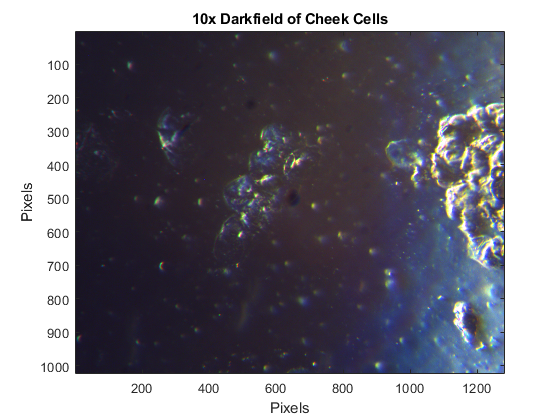
**Figure 1:** Image of cheek cells using 10x Brightfield technique.

****

**Figure 2:** Oblique illumination of cheek cells using 10x Brightfield technique

****

**Figure 3**: Different direction using oblique illumination. The shadows shifted here compared to figure 2.

****

**Figure 4.** Darkfield image of the same cheek cell area us 10x. There still seemed to a bright spot, even after the card was fully inserted.

**Code**

%Part B

bfield = imread('01.bmp');

oblique\_a = imread('02.bmp');

oblique\_b = imread('03.bmp');

dfield = imread('04.bmp');

FigureB1 = figure('Name', '10x Brightfield');

imagesc(bfield);

axis image

title('10x Brightfield of Cheek Cells');

xlabel('Pixels');

ylabel('Pixels');

FigureB2 = figure('Name', '10x Oblique Direction A');

imagesc(oblique\_a);

axis image

title('10x Oblique Illumination of Cheek Cells Direction A');

xlabel('Pixels');

ylabel('Pixels');

FigureB3 = figure('Name', '10x Oblique Direction B');

imagesc(oblique\_b);

axis image

title('10x Oblique Illumination of Cheek Cells Direction B');

xlabel('Pixels');

ylabel('Pixels');

FigureB4 = figure('Name', '10x Darkfield of Cheek Cells');

imagesc(dfield);

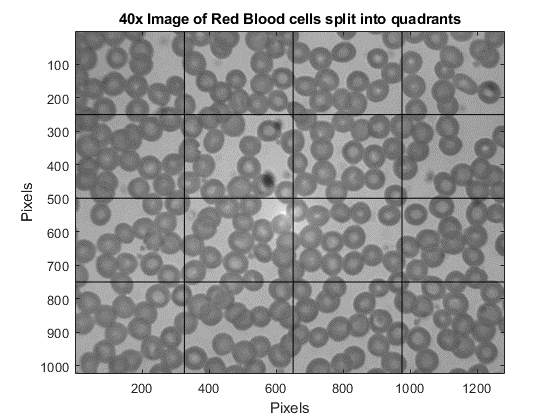
axis image

title('10x Darkfield of Cheek Cells');

xlabel('Pixels');

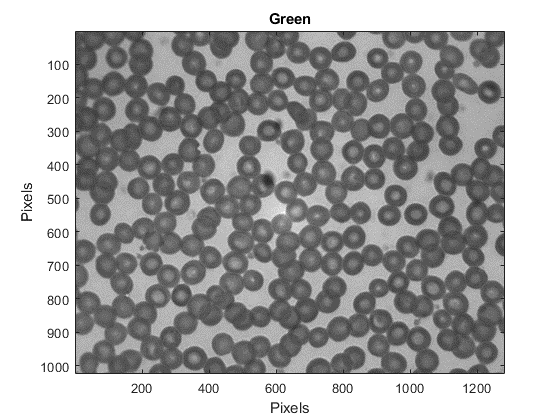
ylabel('Pixels');

**Part C: Contrast Mechanisms based on absorption**

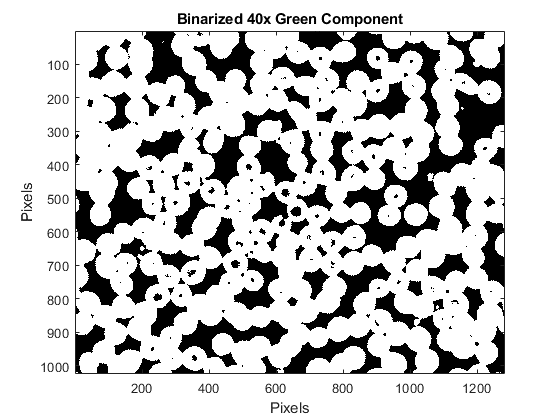
****

**Figure 5:** Red blood cell only images using the 40x objective. There is a black smudge present which is the result of a dirty objective.

There are about 288 red blood cells in the image, given that all the sections on the left have 18 RBC each, extrapolating that to the rest of the image there only seems to be 288 RBCs.

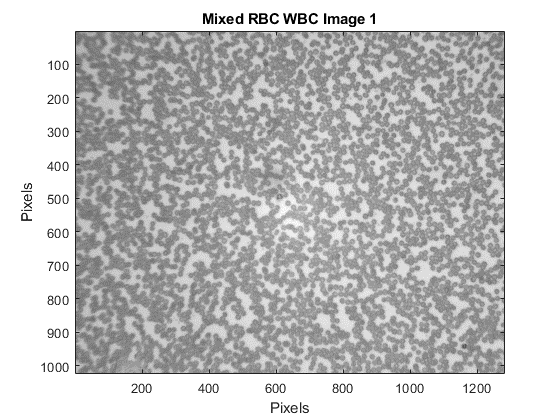


**Figure 6**: Color separated figure 5, displaying only green.



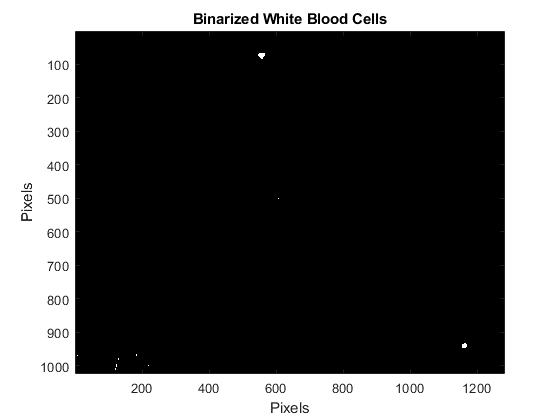
**Figure 7**: Binarized figure 6.

The total pixel area for RBC is 899896 pixels. Given 288 RBC, average area of pixels occupied by single red blood cell (given 40x magnification) is 3124.6 pixels.



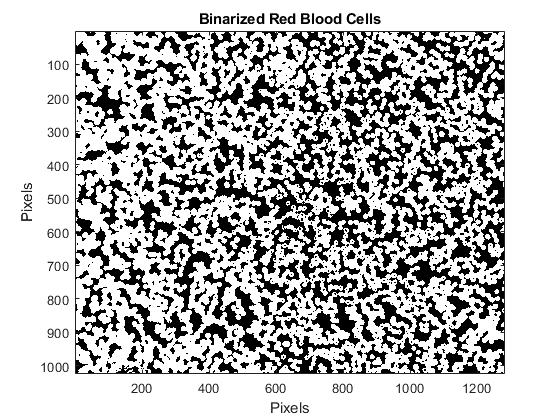
**Figure 8:** Blood sample image containing both RBC and WBC using the 10x objective.

There are seem to be only 2 white blood cells in Figure 8.



**Figure 9**: Binarized figure 8, only white blood cells are apparent.

The total area of white blood cells in figure 9 is 5176 pixels. The average area per white blood cell is 647 pixels (given 8 pixels in the image).



**Figure 10:** Binarized figure 8, only red blood cells are apparent.

The total area of red blood cells is 781773 pixels, which indicates 1000.8 red blood cells.

For the nine mixed images, the average WBC count is 8.5 ± 0.57 while the average RBC count is 1100 ± 164.

The average WBC per square mm is 1.83 \* 104 WBC/mm ± 1.23 \* 103. The average RBC per square mm is 2.03 \* 106 RBC/mm ± 2.92 \*105.

%Part C

RBConly40x = imread('05.bmp');

Mixed\_1 = imread('06\_1.bmp');

Mixed\_2 = imread('06\_2.bmp');

Mixed\_3 = imread('06\_3.bmp');

Mixed\_4 = imread('06\_4.bmp');

Mixed\_5 = imread('06\_5.bmp');

Mixed\_6 = imread('06\_6.bmp');

Mixed\_7 = imread('06\_7.bmp');

Mixed\_8 = imread('06\_8.bmp');

Mixed\_9 = imread('06\_9.bmp');

FigureC1 = figure('Name', '40x RBC only');

Gray40x = 0.2989\*RBConly40x(:,:,1) + 0.5870\*RBConly40x(:,:,2) + 0.1140\*RBConly40x(:,:,3);

colormap(gray)

imagesc(Gray40x);

axis image

title('40x Image of Red Blood cells split into quadrants');

vline(325, 'k') %vline and hline from matlab file exchange

vline(650, 'k')

vline(975, 'k')

hline(250, 'k')

hline(500, 'k')

hline(750, 'k')

xlabel('Pixels');

ylabel('Pixels');

red = squeeze(RBConly40x(:,:,1));

green = squeeze(RBConly40x(:,:,2));

blue = squeeze(RBConly40x(:,:,3));

FigureC2 = figure('Name', 'Red Green Blue Comparison');

% subplot(3,1,1)

% colormap(gray)

% imagesc(red)

% axis image

% title('Red')

% subplot(3,1,2)

colormap(gray) %Green is best

imagesc(green)

axis image

title('Green')

%Green is best

% subplot(3,1,3)

% colormap(gray)

% imagesc(blue)

% axis image

% title('Blue')

xlabel('Pixels');

ylabel('Pixels');

FigureC3 = figure('Name', 'Binarized Green');

binaryg = (green<100);

colormap(gray)

imagesc(binaryg)

axis image

title('Binarized 40x Green Component')

xlabel('Pixels');

ylabel('Pixels');

RBCarea = sum(sum(binaryg));

RBCpixarea = RBCarea/288;

%sequence of 9

Mixed\_1gray = 0.2989\*Mixed\_1(:,:,1) + 0.5870\*Mixed\_1(:,:,2) + 0.1140\*Mixed\_1(:,:,3);

FigureC4 = figure('Name', 'Mixed RBC WBC Image 1');

imagesc(Mixed\_1gray)

colormap(gray)

axis image

title('Mixed RBC WBC Image 1')

xlabel('Pixels');

ylabel('Pixels');

red\_mx1 = squeeze(Mixed\_1(:,:,1));

green\_mx1 = squeeze(Mixed\_1(:,:,2));

blue\_mx1 = squeeze(Mixed\_1(:,:,3));

% FigureC5 = figure('Name', 'Red Green Blue Comparison WBC/RBC');

% subplot(3,1,1)

% colormap(gray)

% imagesc(red) %Red is best for WBC

% axis image

% title('Red')

% subplot(3,1,2)

% colormap(gray) %Green is best for RBC

% imagesc(green\_mx1)

% axis image

% title('Green')

%Green is best

% subplot(3,1,3)

% colormap(gray)

% imagesc(blue)

% axis image

% title('Blue')

% xlabel('Pixels');

% ylabel('Pixels');

FigureC6 = figure('Name', 'Binarized White Blood Cells');

binaryr\_mx1 = (red\_mx1<104);

colormap(gray)

imagesc(binaryr\_mx1)

axis image

title('Binarized White Blood Cells')

xlabel('Pixels');

ylabel('Pixels');

arear\_mx1 = sum(sum(binaryr\_mx1));

wbcparea = arear\_mx1/8

FigureC7 = figure('Name', 'Binarized Red Blood Cells');

binaryg\_mx1 = (green\_mx1<102);

colormap(gray)

imagesc(binaryg\_mx1)

axis image

title('Binarized Red Blood Cells')

xlabel('Pixels');

ylabel('Pixels');

areag\_mx1 = sum(sum(binaryg\_mx1));

rbc\_num = areag\_mx1/(RBCpixarea/4)

%repeat img 1-9

for ii = 1:9

imgname = ['06\_' num2str(ii) '.bmp'];

img\_ii = imread(imgname);

red\_ii = squeeze(img\_ii(:,:,1));

green\_ii = squeeze(img\_ii(:,:,2));

blue\_ii = squeeze(img\_ii(:,:,3));

binaryr\_ii = (red\_ii<104);

binaryg\_ii = (green\_ii<100);

WBCtotarea(ii) = sum(sum(binaryr\_ii));

WBCcount(ii) = WBCtotarea(ii)/wbcparea;

RBCtotarea(ii) = sum(sum(binaryg\_ii));

RBCcount(ii) = RBCtotarea(ii)/(RBCpixarea/4);

end

avgWBCcount = sum(WBCcount)/9

stdevWBCcount = std(WBCcount)

avgRBCcount = sum(RBCcount)/9

stdevRBCcount = std(RBCcount)

%per square mm (0.5 demagnifier) pixel size 3.6

pareamicron\_RBC = (RBCpixarea/4)\*3.6/5

pareamm\_RBC = pareamicron\_RBC \* 10^(-6);

persqmm\_RBC = avgRBCcount/pareamm\_RBC

persqmm\_RBC\_std = stdevRBCcount/pareamm\_RBC

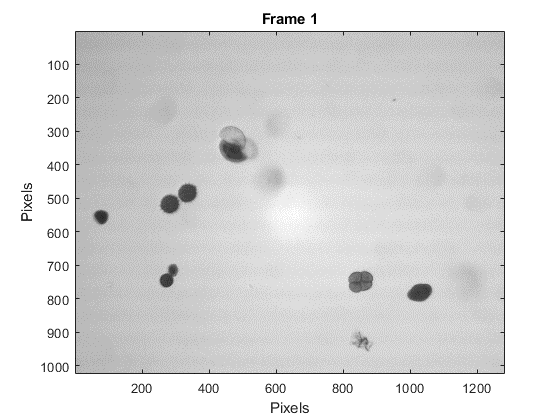
pareamicron\_WBC = (wbcparea)\*3.6/5

pareamm\_WBC = pareamicron\_WBC \* 10^(-6);

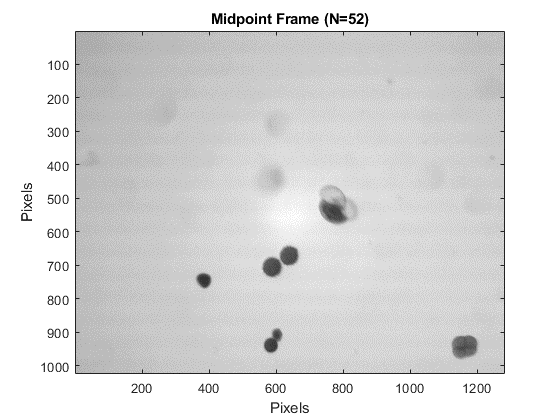
persqmm\_WBC = avgWBCcount/pareamm\_WBC

persqmm\_WBC\_std = stdevWBCcount/pareamm\_WBC

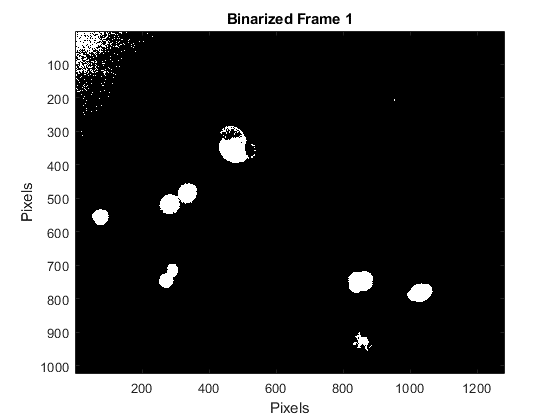
**Part D: Particle Tracking**



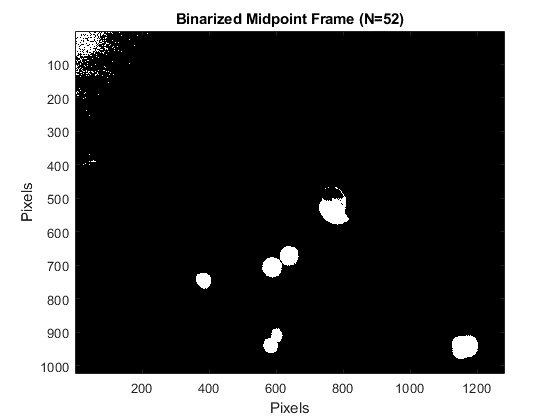
**Figure 11:** Frame 1 of the video of pollen grains.



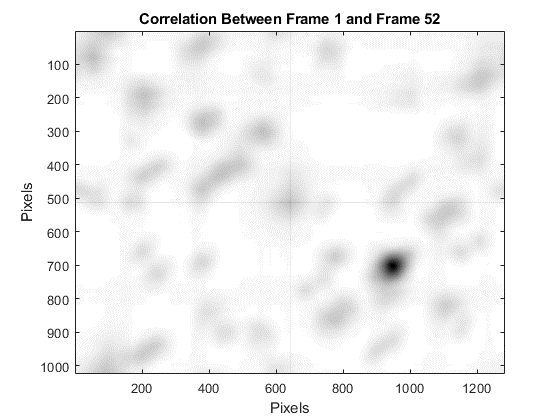
**Figure 12:** Midpoint of the pollen grain video.



**Figure 13:** Binarized Frame 1.

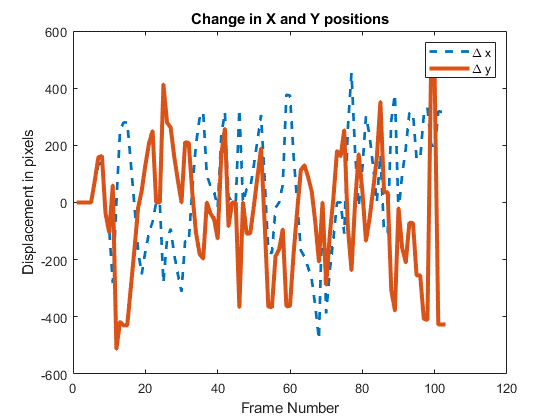


**Figure 14:** Binarized midpoint.



**Figure 15**: Correlation between frame 1 and the midpoint.

The calculated displacement vector of the image is [306, 189] which is in line with the estimation from looking at the image itself.



**Figure 16:** Trajectory of pollen grains throughout the video, relative to frame 1.

%Part D

v = VideoReader('07.avi');

vid = read(v);

[y, x, c, f] = size(vid);

frame1 = squeeze(vid(:,:,:,1));

frame1gray = 0.2989\*frame1(:,:,1)+0.5870\*frame1(:,:,2)+0.1140\*frame1(:,:,3);

frameMid = squeeze(vid(:,:,:,round(f/2)));

frameMidgray = 0.2989\*frameMid(:,:,1)+0.5870\*frameMid(:,:,2)+0.1140\*frameMid(:,:,3);

FigureD1 = figure('Name', 'Frame 1');

colormap(gray)

imagesc(frame1gray)

axis image

title('Frame 1')

xlabel('Pixels')

ylabel('Pixels')

FigureD2 = figure('Name', 'Midpoint Frame');

colormap(gray)

imagesc(frameMidgray)

axis image

title('Midpoint Frame (N=52)')

xlabel('Pixels')

ylabel('Pixels')

redf1 = squeeze(frame1(:,:,1));

greenf1 = squeeze(frame1(:,:,2));

bluef1 = squeeze(frame1(:,:,3));

FigureD3 = figure('Name', 'Binarization Color Chooser');

subplot(3,1,1)

colormap(gray)

imagesc(redf1)

axis image

title('Red')

subplot(3,1,2)

colormap(gray)

imagesc(greenf1)

axis image

title('Green')

subplot(3,1,3)

colormap(gray)

imagesc(bluef1)

axis image

title('Blue')

xlabel('Pixels');

ylabel('Pixels');

%Green is still best

FigureD4 = figure ('Name', 'Binarized Frame 1');

colormap(gray)

bframe1 = (greenf1<100);

imagesc(bframe1)

axis image

title('Binarized Frame 1')

xlabel('Pixels');

ylabel('Pixels');

FigureD5 = figure('Name', 'Binarized Midpoint Frame (N=52)');

colormap(gray)

greenmid = frameMid(:,:,2);

bframeMid = (greenmid<100);

imagesc(bframeMid)

axis image

title('Binarized Midpoint Frame (N=52)')

xlabel('Pixels');

ylabel('Pixels');

fftA = fftshift(fft2(bframe1));

fftB = fftshift(fft2(bframeMid));

fftC = conj(fftA).\* fftB;

imgcorr = ifftshift(ifft2(fftC));

abscor = abs(imgcorr);

FigureD6 = figure('Name', 'Correlation between Frame 1 and Frame 52');

colormap(flipud(gray)) %flipped for easier distinction

imagesc(abscor)

axis image

title('Correlation Between Frame 1 and Frame 52')

xlabel('Pixels');

ylabel('Pixels');

[max\_x, x] = max(max(abscor, [],1));

[max\_y, y] = max(max(abscor,[],2));

[y\_size, x\_size] = size(abscor);

x\_pos = x - (x\_size/2 + 1)

y\_pos = y -(y\_size/2 + 1)

delx = zeros(1,f);

dely = zeros(1,f);

for jj = 1:f

framejj = squeeze(vid(:,:,:,jj));

greenjj = framejj(:,:,2);

bframejj = (greenjj<100);

fftBjj = fftshift(fft2(bframejj));

fftCjj = conj(fftA).\*fftBjj;

imgcorrjj = ifftshift(ifft2(fftCjj));

abscorjj = abs(imgcorrjj);

[max\_x, x] = max(max(abscorjj,[],1));

[max\_y, y] = max(max(abscorjj,[],2));

[y\_size, x\_size] = size(abscorjj);

delx(jj) = x - (x\_size/2 + 1);

dely(jj) = y -(y\_size/2 + 1);

end

FigureD7 = figure('Name', 'Change in x and y positions');

plot(delx, 'LineWidth',2)

hold on

plot(dely, 'LineWidth',3)

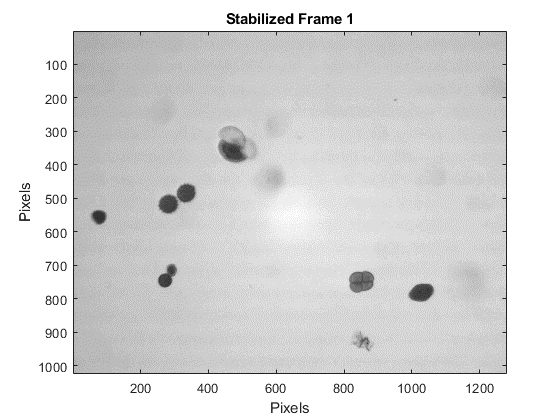
title('Change in X and Y positions')

legend('\Delta x', '\Delta y')

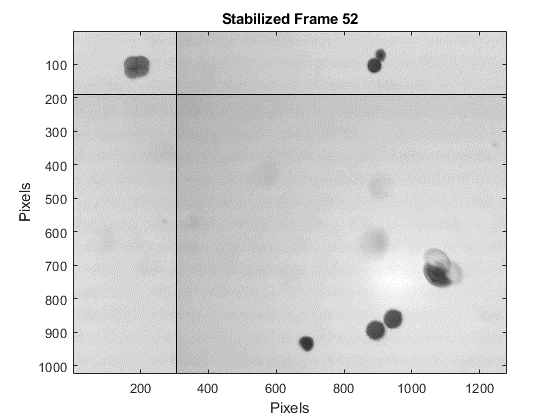
xlabel('Frame Number')

ylabel('Displacement in pixels')

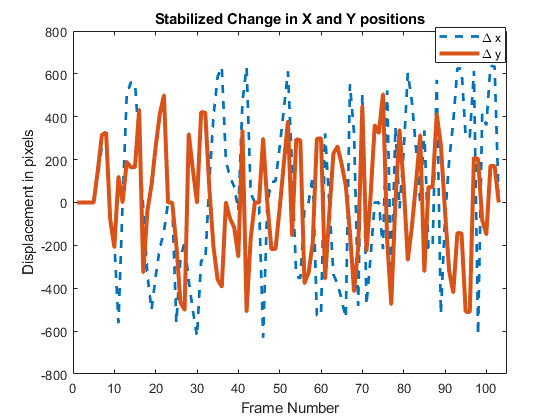
**Part E:**

****

**Figure 17:** Stabilized Frame 1.



**Figure 18:** Stabilized Frame 52.



**Figure 19**: Trajectory of pollen grains from the stabilized video.

**Code**

%% Part E

writer = VideoWriter('new07.avi');

open(writer)

for kk = 1:f-1

framkk = squeeze(vid(:,:,:,kk));

frame\_shift = circshift(framkk, [dely(kk), delx(kk), 0]);

writeVideo(writer,frame\_shift);

end

close(writer)

nv = VideoReader('new07.avi');

nvid = read(nv);

nframe1 = squeeze(nvid(:,:,:,1));

nframe1gray = 0.2989\*nframe1(:,:,1)+0.5870\*nframe1(:,:,2)+0.1140\*nframe1(:,:,3);

nframeMid = squeeze(nvid(:,:,:,round(f/2)));

nframeMidgray = 0.2989\*nframeMid(:,:,1)+0.5870\*nframeMid(:,:,2)+0.1140\*nframeMid(:,:,3);

FigureE1 = figure('Name', 'Stabilized Frame 1');

colormap(gray)

imagesc(nframe1gray)

title('Stabilized Frame 1');

xlabel('Pixels');

ylabel('Pixels');

FigureE2 = figure('Name', 'Stabilized Frame 52');

colormap(gray)

imagesc(nframeMidgray)

title('Stabilized Frame 52');

xlabel('Pixels');

ylabel('Pixels');

greennf1 = squeeze(nframe1(:,:,2));

nbframe1 = (greennf1<100);

greennmid = nframeMid(:,:,2);

nbframeMid = (greennmid<100);

nfftA = fftshift(fft2(nbframe1));

nfftB = fftshift(fft2(nbframeMid));

nfftC = conj(nfftA).\* fftB;

nimgcorr = ifftshift(ifft2(nfftC));

nabscor = abs(nimgcorr);

[nmax\_x, nx] = max(max(nabscor, [],1));

[nmax\_y, ny] = max(max(nabscor,[],2));

[ny\_size, nx\_size] = size(nabscor);

nx\_pos = nx - (x\_size/2 + 1)

ny\_pos = ny -(y\_size/2 + 1)

ndelx = zeros(1,f);

ndely = zeros(1,f);

for zz = 1:f-1

framezz = squeeze(nvid(:,:,:,zz));

greenzz = framezz(:,:,2);

nbframezz = (greenzz<100);

fftBzz = fftshift(fft2(nbframezz));

fftCzz = conj(nfftA).\*fftBzz;

imgcorrzz = ifftshift(ifft2(fftCzz));

abscorzz = abs(imgcorrzz);

[nmax\_x, nx] = max(max(abscorzz,[],1));

[nmax\_y, ny] = max(max(abscorzz,[],2));

[ny\_size, nx\_size] = size(abscorzz);

ndelx(zz) = nx - (nx\_size/2 + 1);

ndely(zz) = ny -(ny\_size/2 + 1);

end

FigureE3 = figure('Name', 'Stabilized Change in x and y positions');

plot(ndelx, 'LineWidth',2)

hold on

plot(ndely, 'LineWidth',3)

title('Stabilized Change in X and Y positions')

legend('\Delta x', '\Delta y')

xlabel('Frame Number')

ylabel('Displacement in pixels')