

Figure 1. Histogram of original image with 3rd order polyfit over threshold

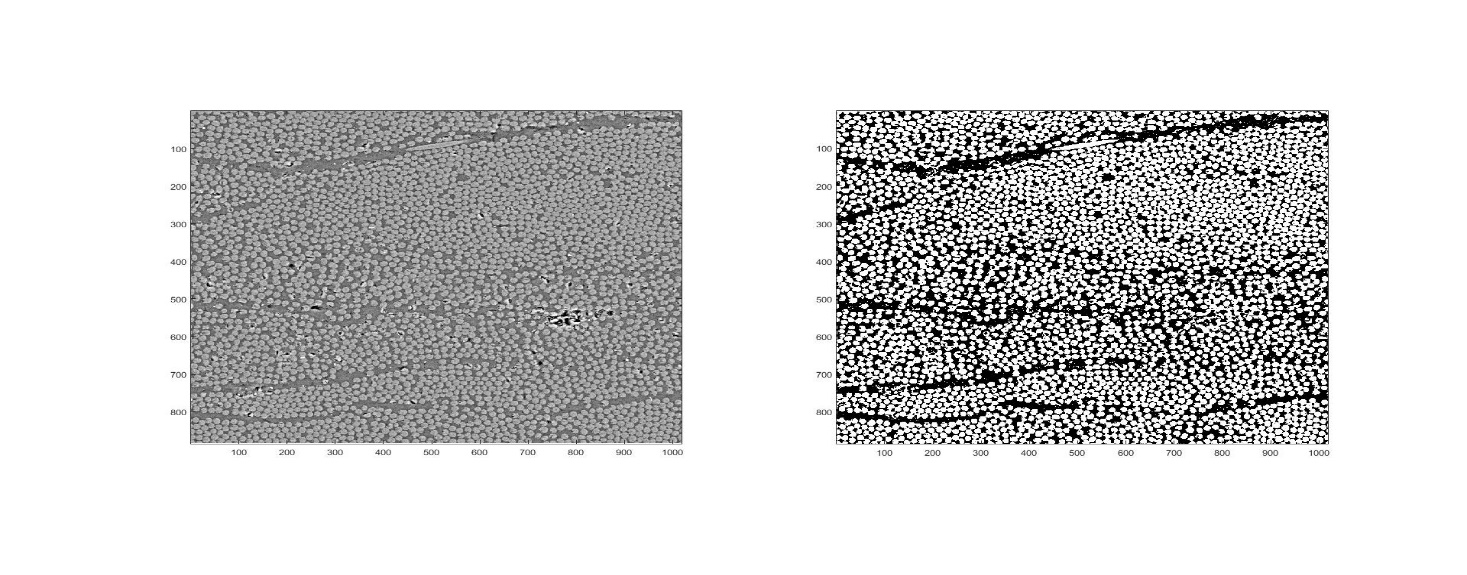


Figure 2. Original SEM image and Processed SEM image. Axes in pixels

imagesize =

495.7660, 430.9950 (microns)

fiber\_dia\_mean =

3.4631 (microns)

function [ proc\_image,FVF ] = HW4\_ressler( img )

%HW4 Reads an image of a composite material and determines the fiber volume

%fraction of the sample.

%% Import the image

image = imread(img);

% Plot the histogram

figure()

imgHist = histogram(image);

hold on

%% Determine Threshold value

% Define manual histogram threshold values

hist\_x\_min = 115;

hist\_x\_max = 150;

% Define x array for polyfitting the dip between histogram peaks

thresh\_fit\_x = hist\_x\_min:hist\_x\_max;

% Fit a ploynomial to the dip, and calculate values for it with polyval

thresh\_fit = polyfit(thresh\_fit\_x,imgHist.Values(thresh\_fit\_x),3);

thresh\_func = polyval(thresh\_fit,thresh\_fit\_x);

% Find the minimum of the poly fit and it's corresponding location in the

% histogram.

[minvalue,minlocation] = min(thresh\_func);

threshold = minlocation + hist\_x\_min;

% Plot the polynomial values over the histogram

plot(thresh\_fit\_x,thresh\_func);

hold off

%% Maximize contrast

% Define image dimensions

[height,width] = size(image);

% Preallocate memory for the processed image

proc\_image = image;

% Define high and low threshold values

highthreshold = 150;

lowthreshold = 125;

% Loop over each pixel to maximize contrast.

for x = 1:width

for y = 1:height

%If the pixel is above the high threshold value...

if image(y,x) > highthreshold

%... make the value in the processed image 255

proc\_image(y,x) = 255;

% Otherwise if the pixel value is less than the low threshold...

elseif image(y,x) < lowthreshold

%... make the value in the processed image 0

proc\_image(y,x) = 0;

end

end

end

% This is an intermediate step to show the segmented image before and

% after clean-up. Uncomment these lines and the line below the next for

% loop to show the figure:

% figure()

% subplot(1,2,1), imshow(proc\_image)

% hold on

% Loop to clean up ambiguous pixels

for x = 1:width

for y = 1:height

% Try to determine fiber/matrix from surrounding pixels if the

% pixel is between the low and high thresholds.

if image(y,x) >= lowthreshold && image(y,x) <= highthreshold

% only check pixels inside the edge (to avoid accessing pixels

% outside of the image)

if x>1 && y>1 && x<width-1 && y<height-1

% Define the pixels around the ambiguous pixel

checkpix = [image(y-1,x-1) image(y-1,x) image(y-1,x+1)...

image(y,x-1) image(y,x+1)...

image(y+1,x-1) image(y+1,x) image(y+1,x+1)];

% Loop to check each pixel and define a probable value

% Number of surrounding pixels that are high or low

highpix = 0;

lowpix = 0;

% Loop to check each surrounding pixel and track number of

% high/low pixels

for pix = 1:8

if checkpix(pix) > highthreshold

highpix = highpix +1;

elseif checkpix(pix) < lowthreshold

lowpix = lowpix +1;

end

end

% If there are more high pixels surrounding the ambiguous one,

% it's probably high

if highpix > lowpix

proc\_image(y,x) = 255;

% Otherwise there are more low pixels surrounding the ambiguous

% one, and it's probably low

else

proc\_image(y,x) = 0;

end

end

end

end

end

%% Determine FVF

% calculate a histogram for the processed image

proc\_hist = histogram(proc\_image);

% grab the number of fiber pixels from the bin that includes 255

fibers = proc\_hist.Values(51);

% grab the number of matrix pixels from the bin that includes 0

matrix = proc\_hist.Values(1);

% Calculate fiber volume fraction.

FVF = fibers/(fibers+matrix);

%% Display image with original image

% Create a new figure

figure()

% Create subplots and put each image into one of the slots

subplot(1,2,1), subimage(image)

subplot(1,2,2), subimage(proc\_image)

%% Determine size of the image

pixel\_pitch = 487e-3; % in microns

imagesize = [pixel\_pitch\*width, pixel\_pitch\*height]

%% Determine average fiber diameter in micron

% Create and array of fibers to calculate diameter

fibers = [348 201; 360 201; 346 216; 359 216; 968 227; 980 227; 475 720; 475 736; 188 766; 188 777];

% Calculate fiber diameters

for i = 1:2:10

% Take the max of the subtraction between pixels (one should always be

% zero, from horizontal / vertical measurement)

fiber\_dia\_pix(i) = max(fibers(i+1,:)-fibers(i,:));

end

% calulate mean fiber diameter

fiber\_dia\_mean = mean(fiber\_dia\_pix) \* pixel\_pitch

end