

---

```
%Dr. Sankalp Bhan
%ESE 448

%References:
% #1. Digital Image Processing with Matlab: Gonzalez, Woods, Eddins
% #2. https://www.mathworks.com/help/images/hough-
transform.html#buh9ylp-26

% load up some image data
%load('ThirdTurnSuccess.mat');
```

## Load an Image

```
figure;
idx = 28;
% R = rt_colorR.signals.values(:,:,idx);
% G = rt_colorG.signals.values(:,:,idx);
% B = rt_colorB.signals.values(:,:,idx);
R = rt_R.signals.values(:,:,idx);
G = rt_G.signals.values(:,:,idx);
B = rt_B.signals.values(:,:,idx);

picture = cat(3,R,G,B);
greyImage = rgb2gray(picture);

imshow(picture)

% apply an intensity transformation (to be discussed later)
greyImage = imcomplement(greyImage);
% (x-1,y-1) (x-1,y ) (x-1,y+1)
% (x,y-)    (x,y)    (x,y+1)
% (x+1,y-1) (x+1,y)  (x+1,y+1)
% w(-1,-1) w(-1,0) w(-1,+1)
% w(0,-1) w(0,0) w(0,+1)
% w(+1,-1) w(1,0) w(+1,+1)
% 1/9 1/9 1/9
% 1/9 1/9 1/9 = M
% 1/9 1/9 1/9
N=3
w = ones(N)/N^2;
filtered_image = imfilter(greyImage,w);
figure
subplot(1,2,1); imshow(greyImage);

subplot(1,2,2); imshow(filtered_image)
adjGreyImage = imadjust(filtered_image,[0 0.15],[0,1]);

figure
imhist(greyImage);
hold on;
imhist(filtered_image)
```

---

```

imhist(adjGreyImage)

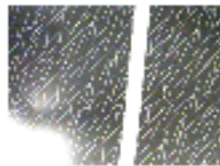
figure;
imshow(adjGreyImage)
% detect horizontal lines
w_h = [-1 -1 -1 ;...
        2 2 2 ;...
        -1 -1 -1 ] ;
w_v = [-1 2 -1 ; -1 2 -1 ; -1 2 -1];
N = 15;
w_v_15 = [-N*ones(N+2,1) 2*ones(N+2,N) -N*ones(N+2,1)];
h = imfilter(adjGreyImage , w_h) ;
v = imfilter(adjGreyImage, w_v);
v_15 = imfilter(adjGreyImage, w_v_15);

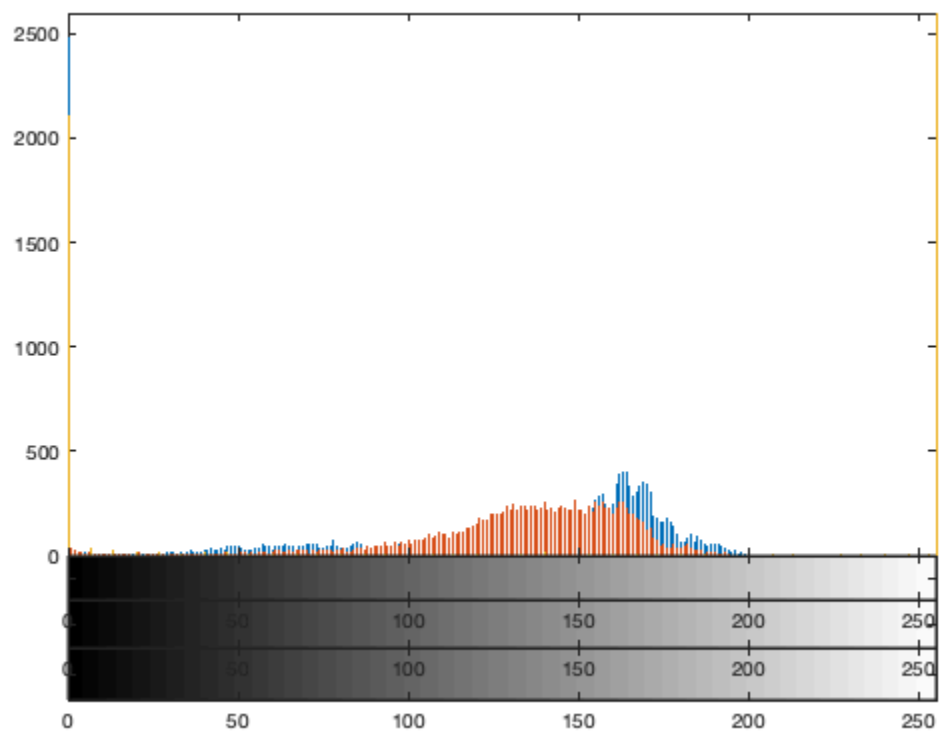
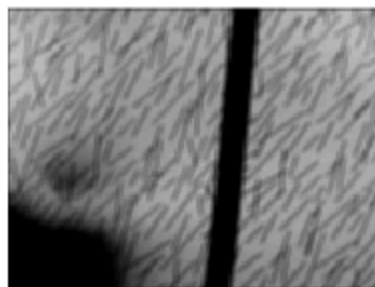
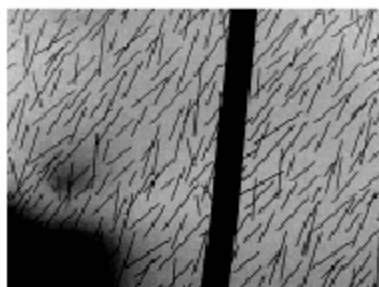
figure
subplot(131)
imshow ( h )
subplot(132)
imshow(v);
subplot(133)
imshow(v_15)
figure
sob_thresh = [0.3]; % try 0.3
can_thres = [1/2, 7/8]; % try [1/2, 7/8] with lo/hi
can_sigma = [];

```

$N =$

3







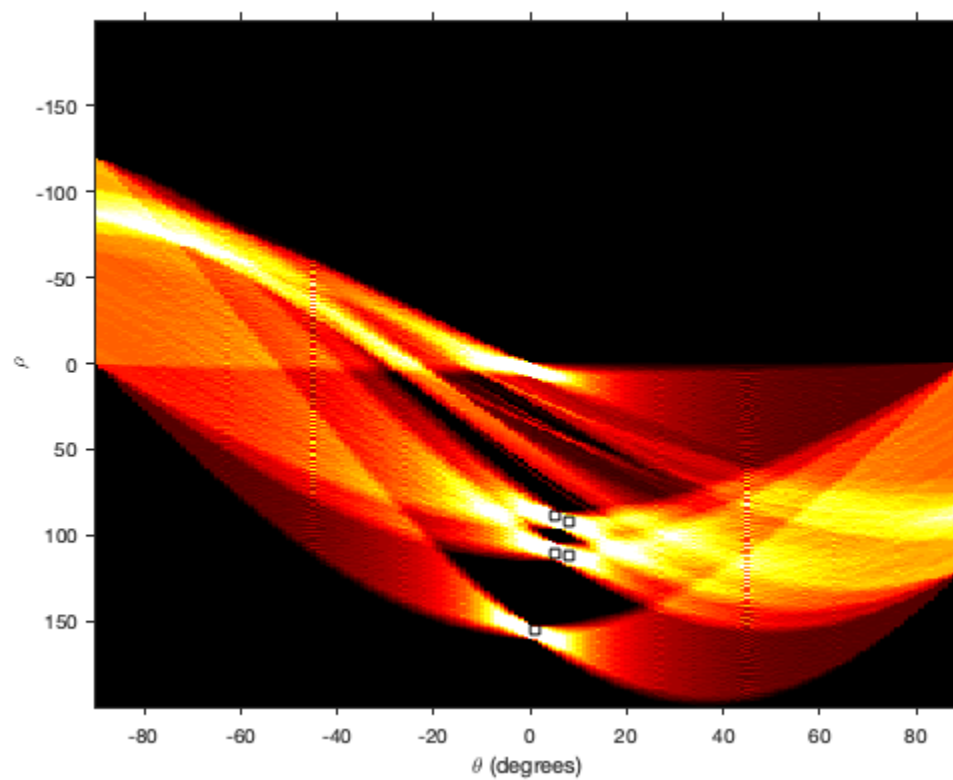
---

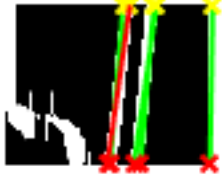
## Grab the edges

```
v_151 = edge(adjGreyImage, 'sobel', 'vertical', sob_thresh);
v_152 = edge(adjGreyImage, 'canny', can_thres);
subplot(1,2,1); imshow(v_151); subplot(1,2,2); imshow(v_152)
[H,theta,rho] = hough(v_15);
figure
imshow(imadjust(rescale(H)),[],...
       'XData',theta,...
       'YData',rho,...
       'InitialMagnification','fit');
xlabel('\theta (degrees)')
ylabel('\rho')
axis on
axis normal
hold on
colormap(gca,hot)
P = houghpeaks(H,5, 'threshold',ceil(0.3*max(H(:)))));
x = theta(P(:,2));
y = rho(P(:,1));
plot(x,y, 's', 'color', 'black');
lines = houghlines(v_15,theta,rho,P, 'FillGap',5, 'MinLength',7);
figure, imshow(v_15), hold on
max_len = 0;
for k = 1:length(lines)
    xy = [lines(k).point1; lines(k).point2];
    plot(xy(:,1),xy(:,2), 'LineWidth',2, 'Color', 'green');

    % Plot beginnings and ends of lines
    plot(xy(1,1),xy(1,2), 'x', 'LineWidth',2, 'Color', 'yellow');
    plot(xy(2,1),xy(2,2), 'x', 'LineWidth',2, 'Color', 'red');

    % Determine the endpoints of the longest line segment
    len = norm(lines(k).point1 - lines(k).point2);
    if ( len > max_len)
        max_len = len;
        xy_long = xy;
    end
end
% highlight the longest line segment
plot(xy_long(:,1),xy_long(:,2), 'LineWidth',2, 'Color', 'red');
```





*Published with MATLAB® R2021a*