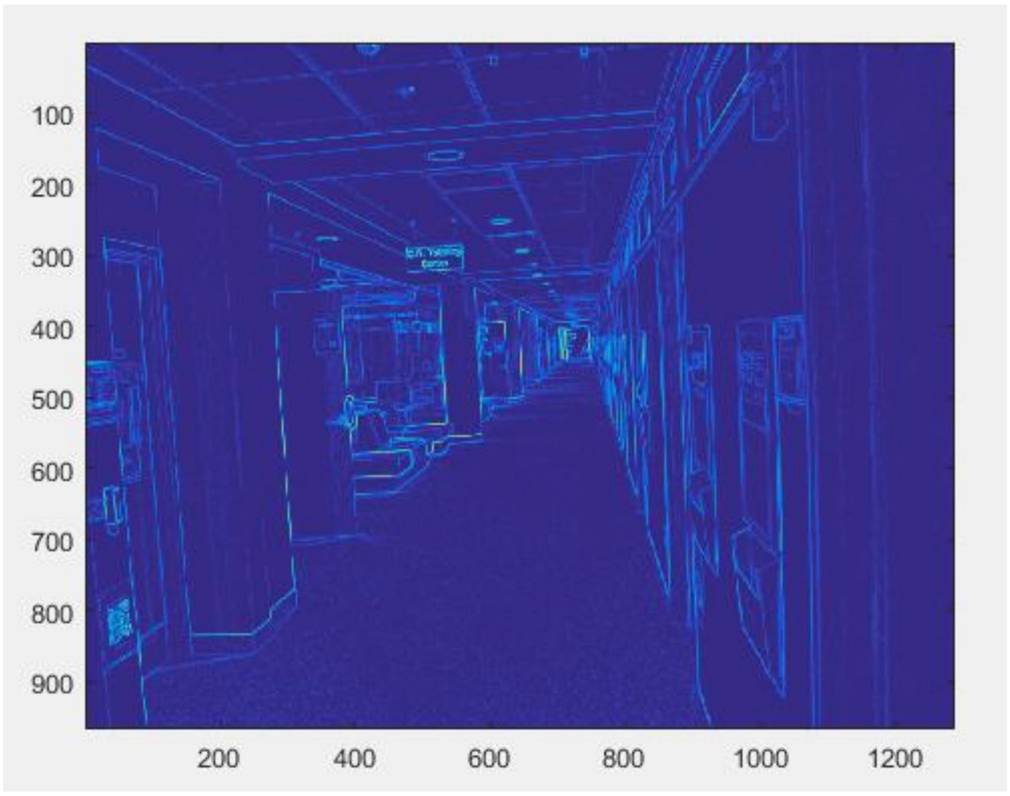
HomeWork 8 Deepak Sharma

Step 0. Apply Gaussian Filter (Abort this step if you are planing to use canny Edge Detector in next step)

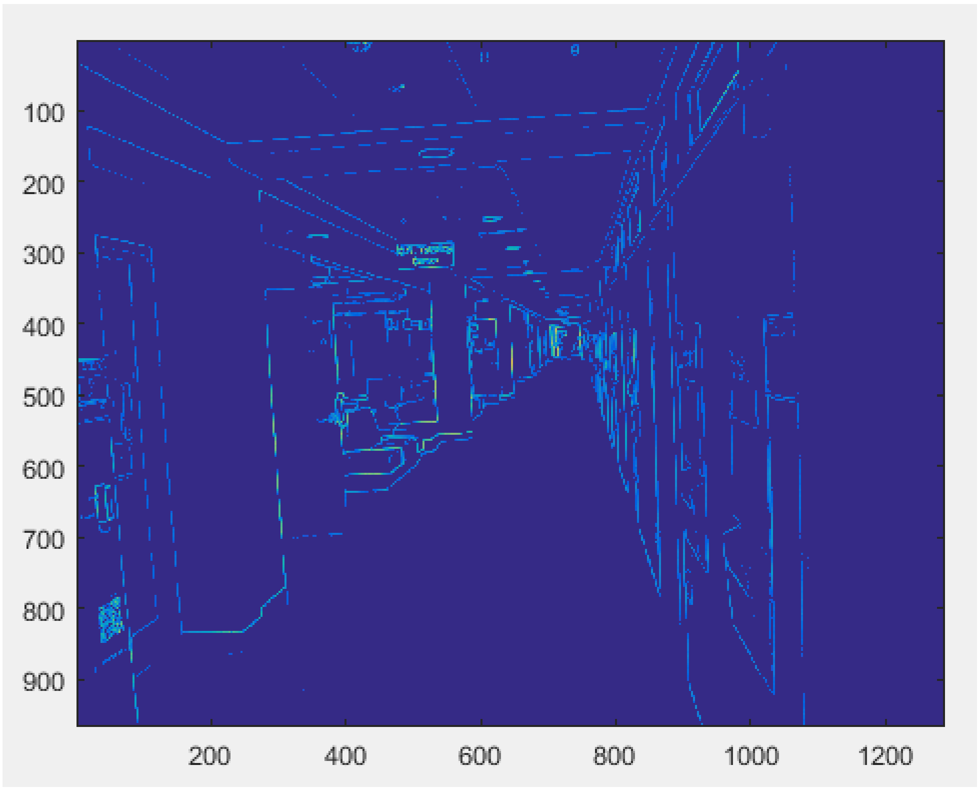
Step1. Calculated Edge Magnitude and Edge Direction using sobal edge detector:

```
function [imgEdge,edgeAngle]=Sobel(img)
    img=im2double(img);
    sobelX= [1 2 1;
             0 0 0;
            -1 -2 -1];
    sobelY= [1 0 -1;
             2 0 -2;
             1 0 -1];

    imgEdgeX=imfilter(img,sobelX,'same');
    imgEdgeY=imfilter(img,sobelY,'same');
    imgEdge = (imgEdgeX.^2+imgEdgeY.^2).^(1/2);
    edgeAngle=atan(imgEdgeY./imgEdgeX);
end
```



Step 2 Kept only top 20% mangitude of edges, set rest to 0.

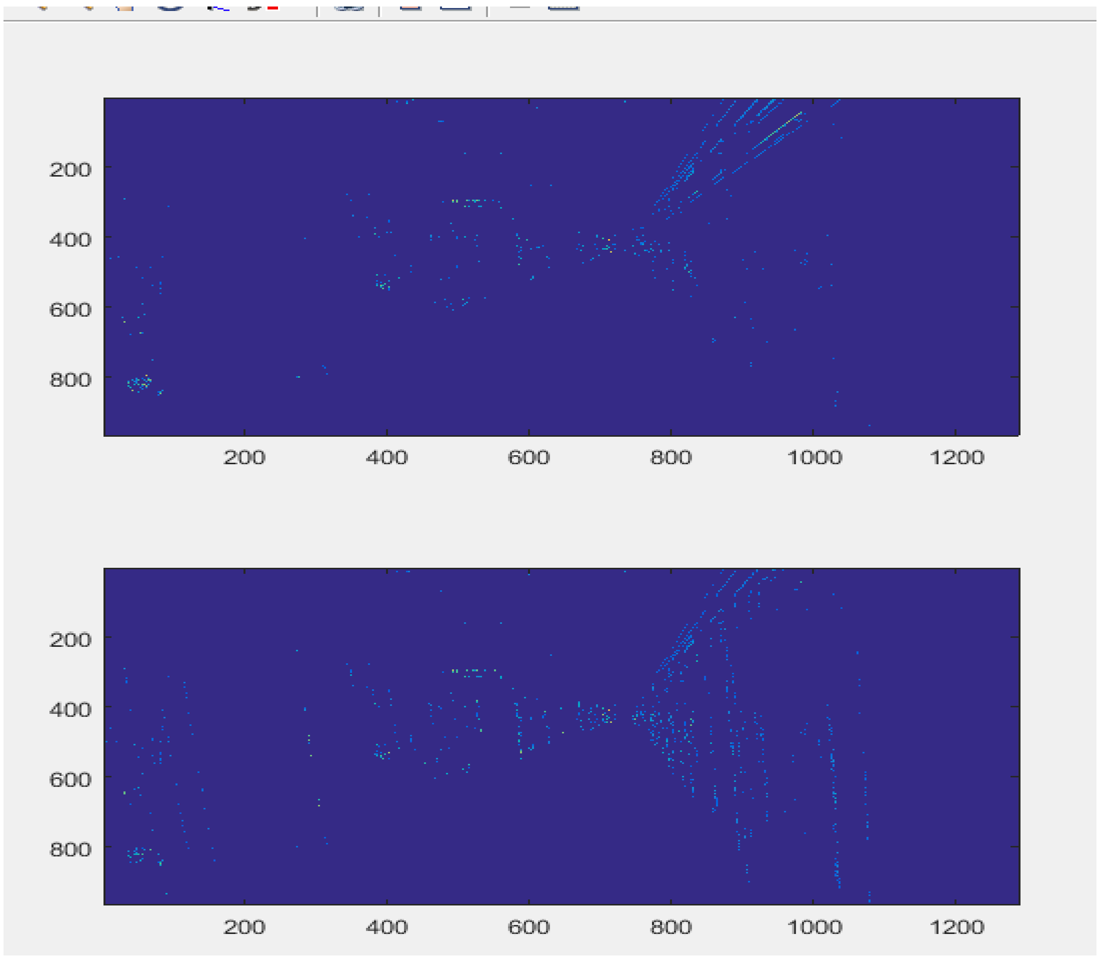
```
function mag = preProcessing_mag(mag)
mag_col=mag(:);
mag_col_sorted=sort(mag_col);
top_value_index=round(size(mag_col_sorted,1)*.20);
thrshold=mag_col_sorted(size(mag_col_sorted,1)-top_value_index);

for counter=1:size(mag_col,1)
    if mag_col(counter)<thrshold
        mag_col(counter)=0;
    end
end
mag=reshape(mag_col,size(mag));
```



Step3 For various interval of angels value, repeated below steps.

1. Removed all the edges which do not fell whiten the tolerance range (20) for specific angle.

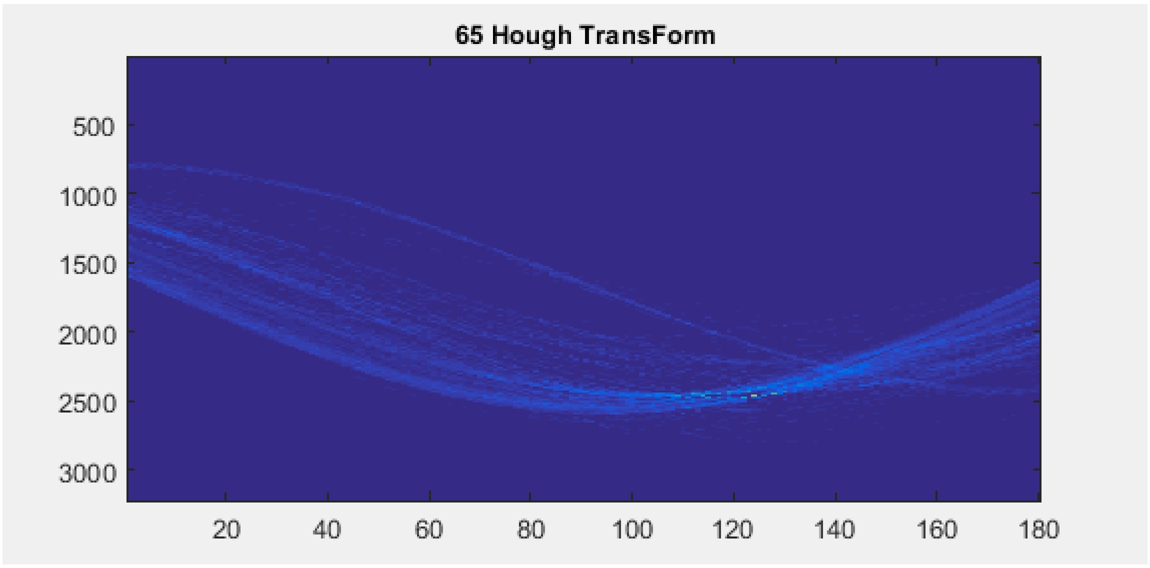
```
function mag_angle = preProcessing_angle(mag,edge_angle,engles)
     new_mag=zeros(size(mag));
     for row=1:size(edge_angle,1)
         for col=1:size(edge_angle,2)
             if (edge_angle(row,col)>=engles-20) &&
(edge_angle(row,col)<=engles+20)
                 new_mag(row,col)=mag(row,col);
             end

         end
     end

mag_angle=new_mag;
end
```



2. Calculated Hough Transform for these edges and find the set of row and theta(Peaks) for which maximum value recorded in hough transform. Each set of row and Theta

represent a line(or multiple disconnected line). Find these lines using houghlines method.

```
[H,theta,rho] = hough(mag_at_angle);
peaks = houghpeaks(H,5, 'threshold',ceil(0.3*max(H(:))));
lines=houghlines(mag_at_angle,theta,rho,peaks,'FillGap',35,'MinLength',20);
```



3. Find the equation of the line and based on distance of each coordinate of the image from this line, accumulated votes for in a 2D matrix (Can use MashGrid for generating coordinate points).

```
function [m,c]=line_equation(point)
   firstPoint=point(1,:);
   secondPoint=point(2,:);
   m=(firstPoint(2)-secondPoint(2))/(firstPoint(1)-
   secondPoint(1));
   c=firstPoint(2)- m*firstPoint(1);
end
```

4. All the points which were within the distnace of 1.5 pixel from the line received the votes perpotional to the length of the line.
   Pont (x1,y1) 's distance from line y-mx-c=0 is |y1-mx1-c|/(1+m^2)^(1/2)

```
            [m,c]=line_equation(point);
             line_length=ceil( CalcDistance(point(1,:),point(2,:))*10);
             for row=1:size(mag,1)
                 for col=1:size(mag,2)
                      if abs(row-m*col-c) <1.5
vote_accumulator(row,col)=vote_accumulator(row,col)+line_length;
                      end
                 end
```

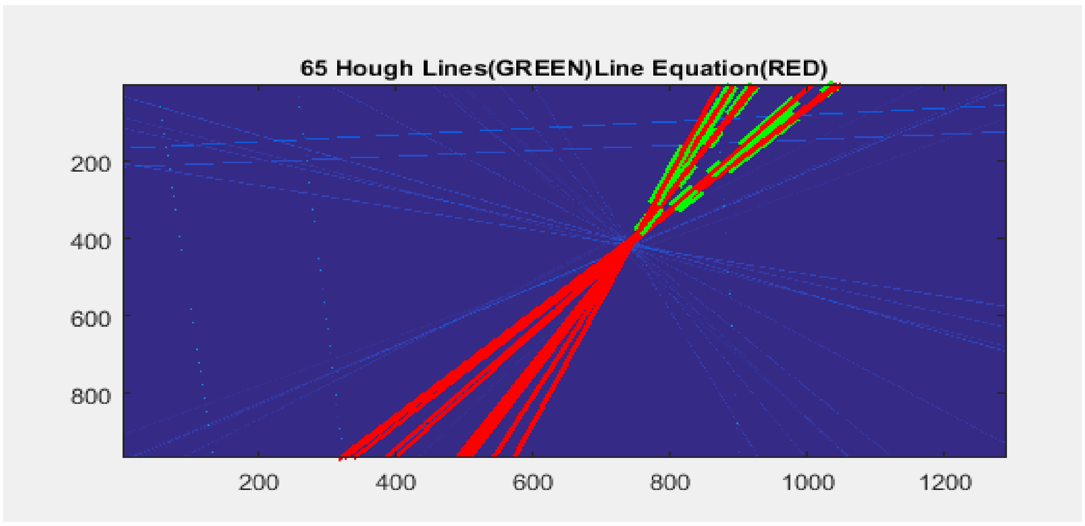
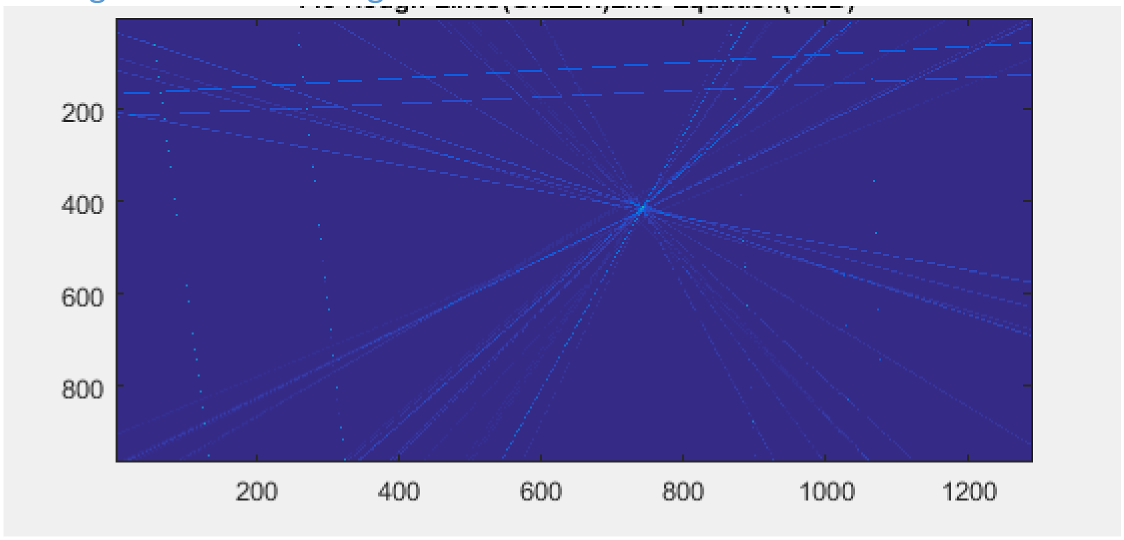


Fig Vote Accumulation for 65 degree

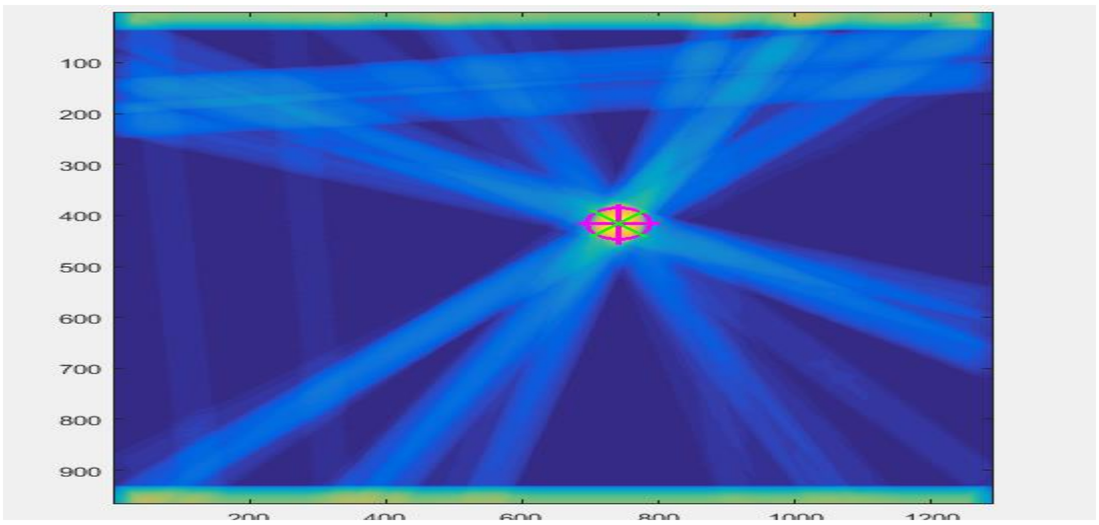
4. Repeated step3 for all Angle Intervals, and recorded cumulative votes by adding votes of each Angle intervel.



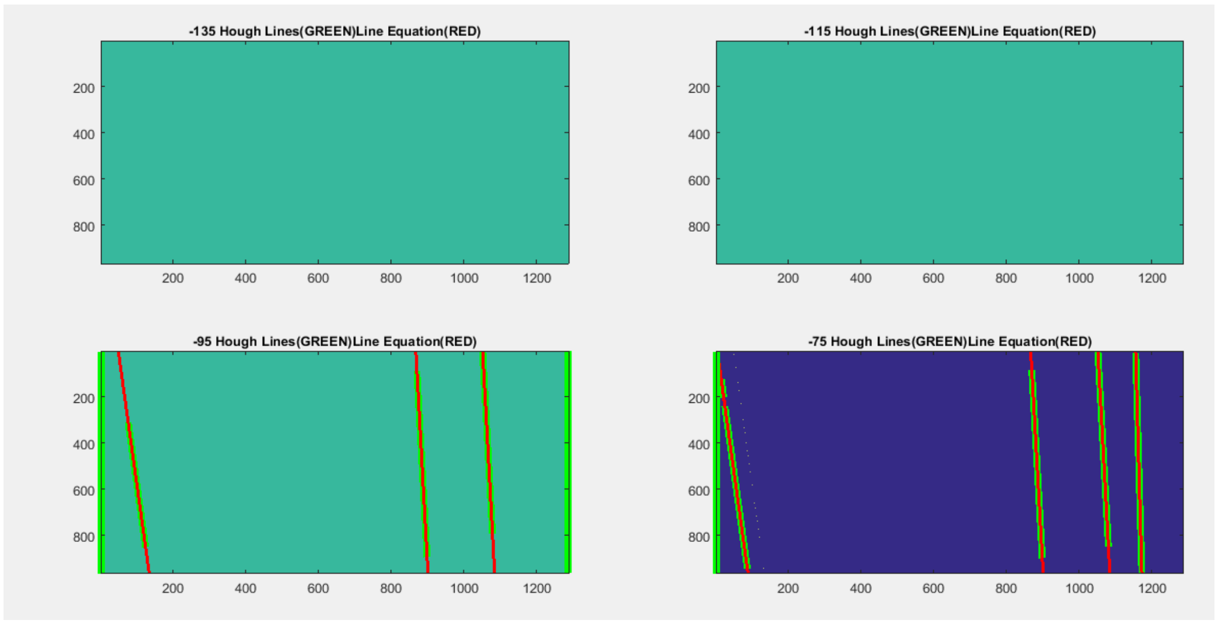
5. Applied disk fiter on cummulative acumulation of vote to fix aliasing effect by smearing the votes.

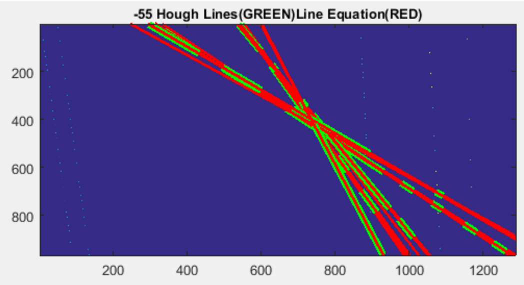
```
gaussian=fspecial('disk',30);
vote_accumulator=imfilter(vote_accumulator,gaussian);
```

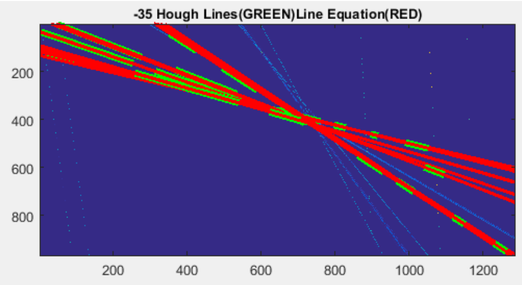
6. Find The coordinate which received maximum vote and assigne Vansihing point to the coordinate.

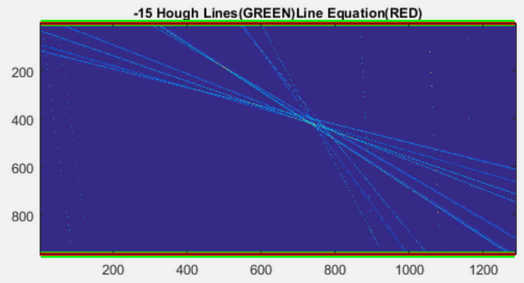


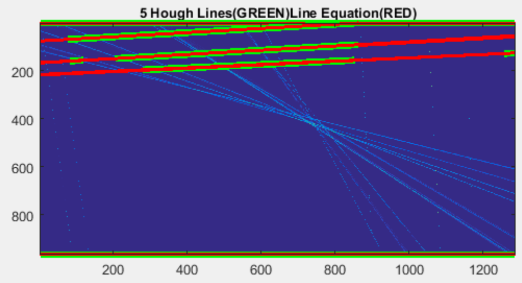
Angle By angle Accumulation of votes:-

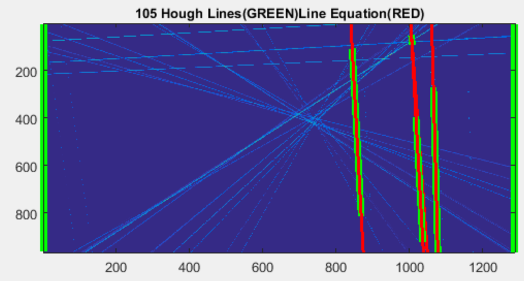


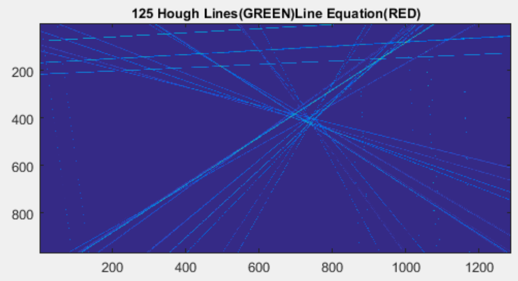

-55 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

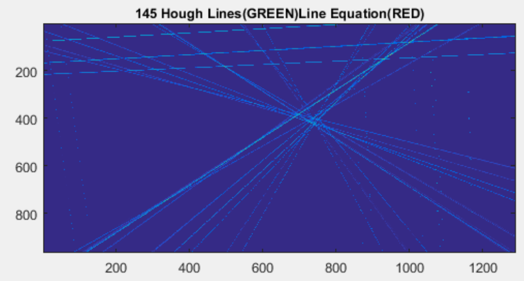

-35 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

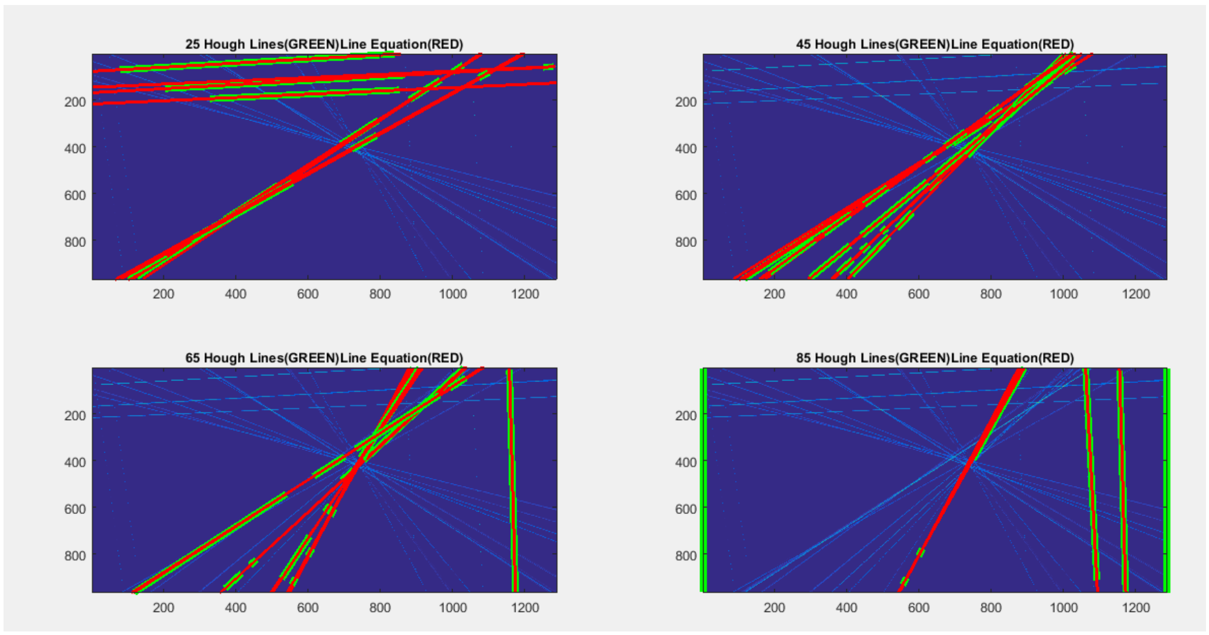

-15 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200


5 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

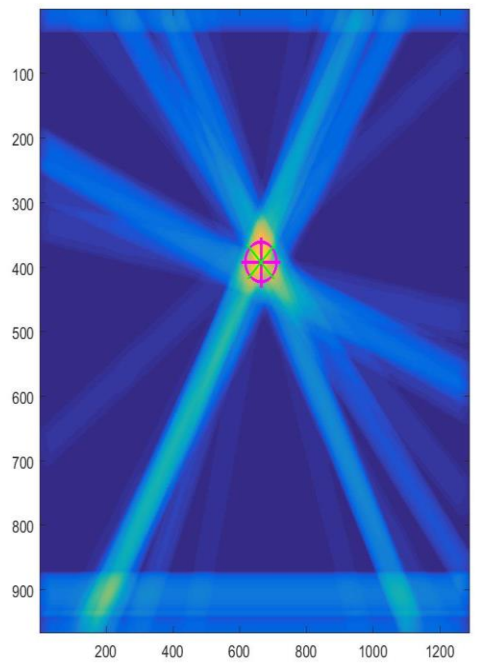

105 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

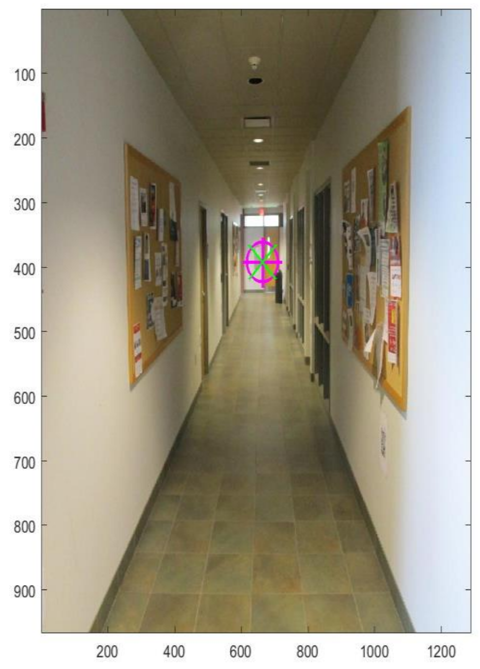

125 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

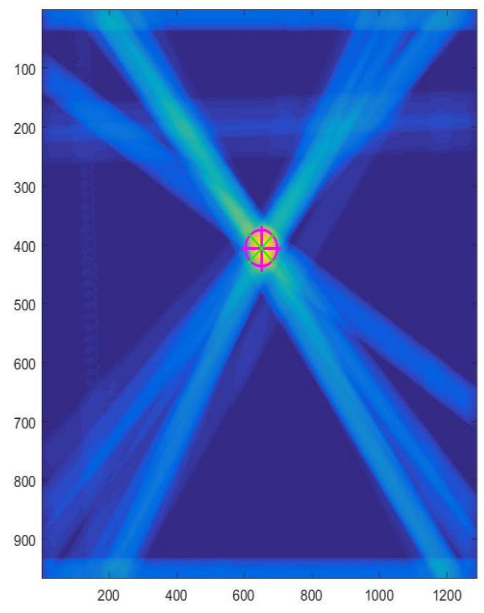

145 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
200
400
600
800
1000
1200

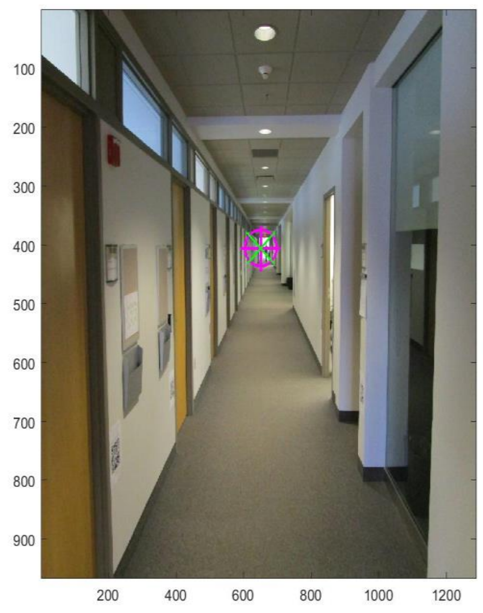

25 Hough Lines(GREEN)Line Equation(RED)
45 Hough Lines(GREEN)Line Equation(RED)
65 Hough Lines(GREEN)Line Equation(RED)
85 Hough Lines(GREEN)Line Equation(RED)
200
400
600
800
1000
1200

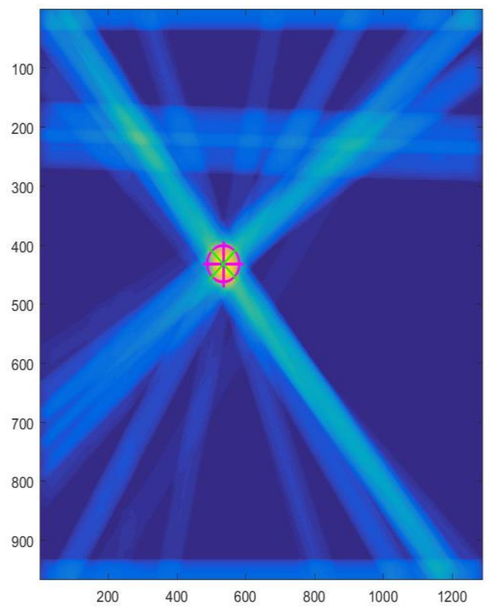
Outputs:-

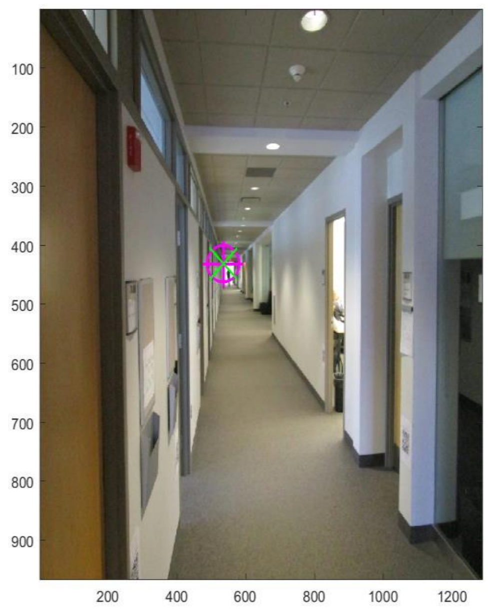

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

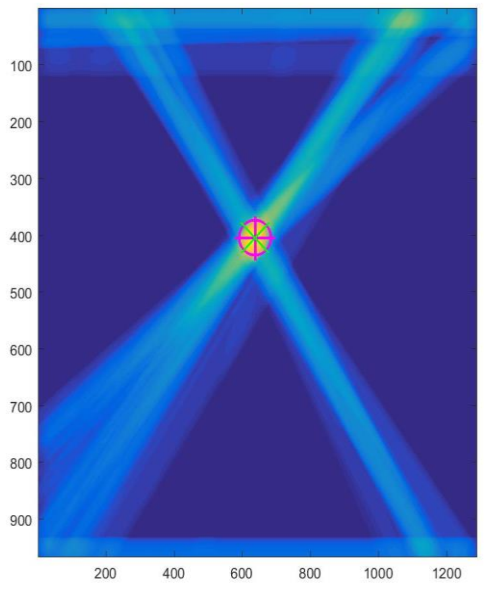

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

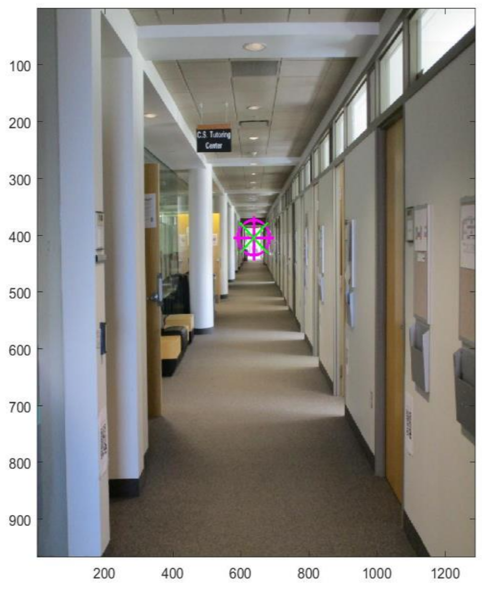

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

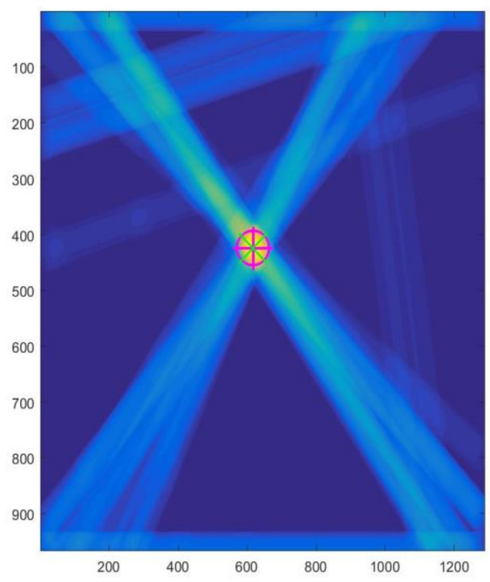

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

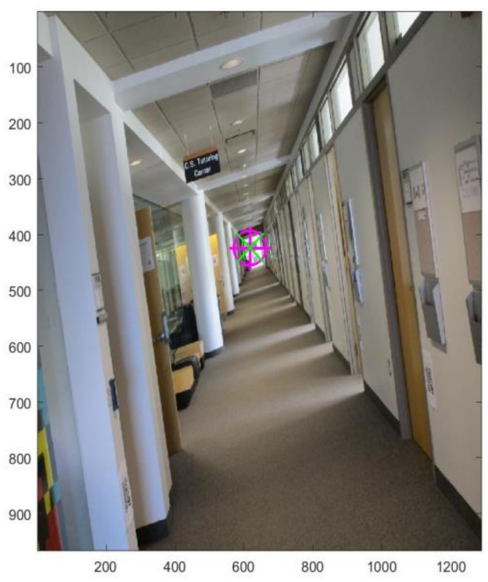

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

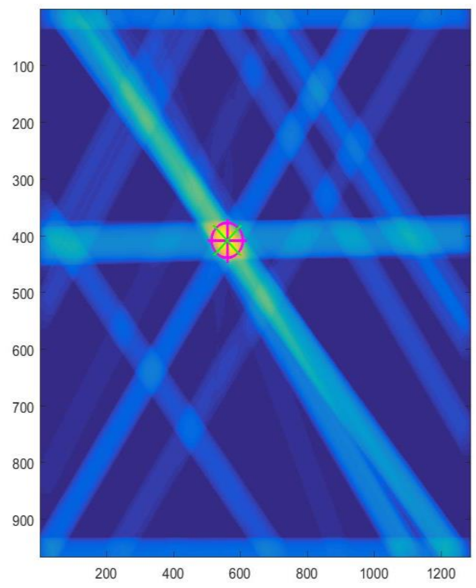

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

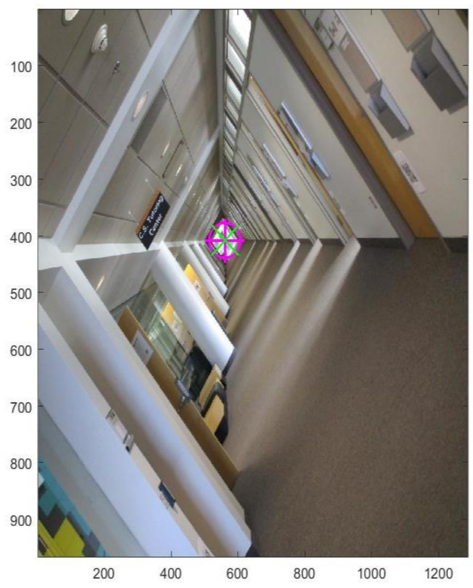

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

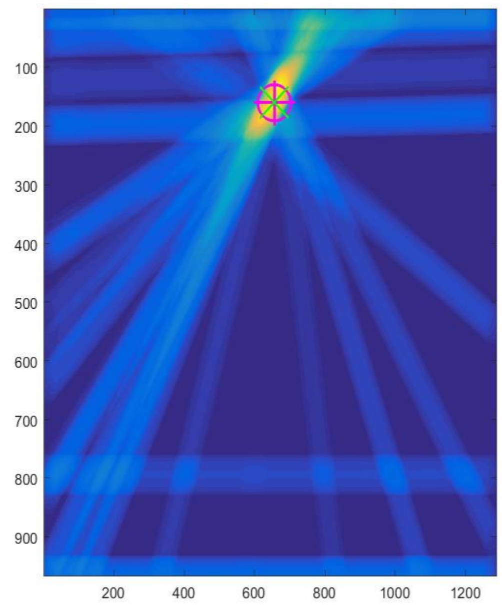

C.S. Tutoring Center
100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200


100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

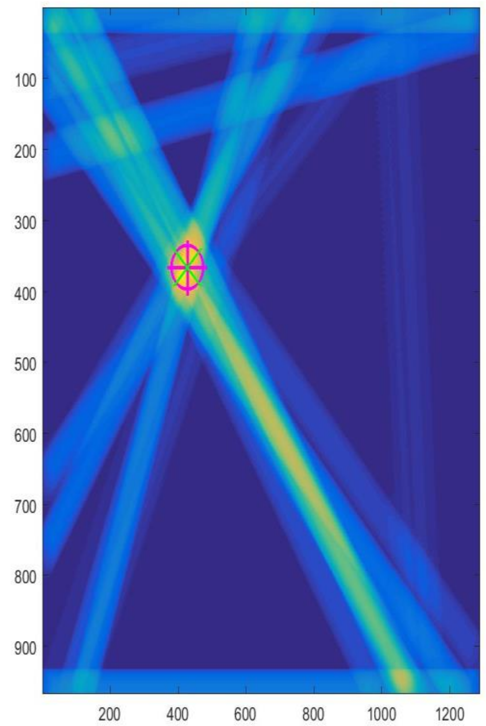

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200


100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

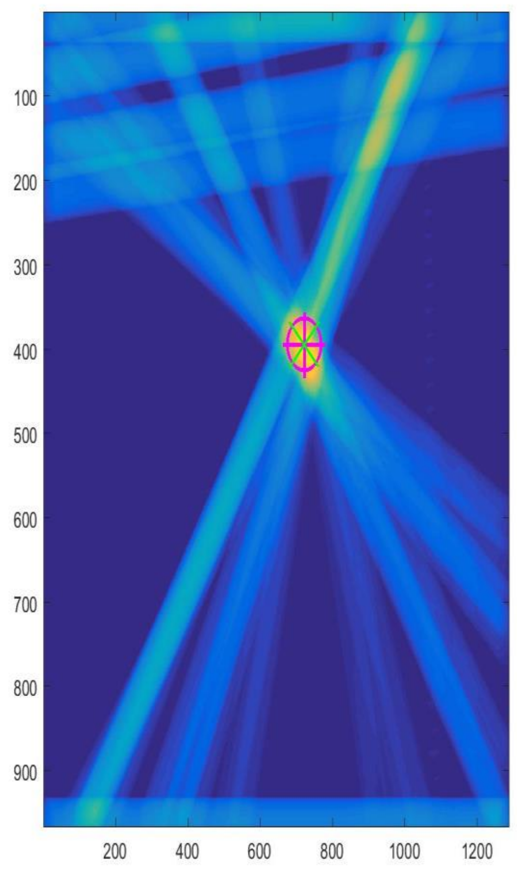

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200


100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200


100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200


100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200

100
200
300
400
500
600
700
800
900
200
400
600
800
1000
1200