I. 2: Theory Questions

A. *Q2.1 Show that if you use the line equation, each image point (x; y) results in a sinusoid in (ρ; θ) Hough space. Relate the amplitude and phase of the sinusoid to the point (x; y)*

Answer:

$$\rho = x * cos(\theta) + y * sin(\theta) \tag{1}$$

Let x_i and y_i be a point on the line in the image space which maps to a sinusoid in the parameter space, then

$$\rho = x_i * cos(\theta) + y_i * sin(\theta) \tag{2}$$

Multiply and divide Equation(2) with distance of the point from the origin

$$\sqrt{x_i^2 * + y_i^2} \tag{3}$$

Equation(2) becomes

$$\rho = \sqrt{x_i^2 * + y_i^2} * (\frac{x_i * cos(\theta)}{\sqrt{x_i^2 * + y_i^2}} + \frac{y_i * sin(\theta)}{\sqrt{x_i^2 * + y_i^2}}) \tag{4}$$

Using Trignometry

$$\rho = \sqrt{x_i^2 * + y_i^2} * (sin(\alpha) * cos(\theta) + cos(\alpha) * sin(\theta) \tag{5}$$

where,

$$sin(\alpha) = \frac{x_i}{\sqrt{x_i^2 * + y_i^2}} \tag{6}$$

and

$$cos(\alpha) = \frac{y_i}{\sqrt{x_i^2 * + y_i^2}} \tag{7}$$

Equation(4) becomes

$$\rho = \sqrt{x_i^2 * + y_i^2} * (sin(\alpha + \theta)) \tag{8}$$

Now let

$$\sqrt{x_i^2 * + y_i^2} = \beta \tag{9}$$

then

$$\rho = \beta * (sin(\alpha + \theta)) \tag{10}$$

where β is amplitude of the sine wave and α is the phase shift

B. *Q2.2 Why do we parameterize the line in terms of ρ; θ instead of slope and intercept (m; c)? Express the slope and intercept in terms of rho and theta.*

Answer: If we do parameterize the line in slope intercept form the accumulator array becomes very large what I mean by that is the range of the slope can be from negative infinity to positive infinity and hence there is no bound on the accumulator array. Additionally since the slope of a straight vertical line is infinity it cannot be represented in slope intercept form (Veritcal lines give rise to unbounded values in the parameter space)

Normal form of a line is given by,

$$\rho = x_i * cos(\theta) + y_i * sin(\theta) \tag{11}$$

Slope intercept form is given by

$$y = m * x + c \tag{12}$$

Equating 11 and 12, we get

$$y = \frac{\rho}{sin(\theta)} - x * \frac{cos(\theta)}{sin(\theta)} \tag{13}$$

Comparing equation 13 with 12

$$m = -\frac{cos(\theta)}{sin(\theta)} \tag{14}$$

and

$$c = \frac{\rho}{sin(\theta)} \tag{15}$$

C. *Q2.3 Assuming that the image points (x; y) are in an image of width W and height H , what is the maximum absolute value of ρ and what is the range of θ?*

Answer: The maximum absolute value of ρ is $\sqrt{W^2 + H^2}$, the range of θ is [0:360]

D. *Q2.4 For points (10; 10), (15; 15) and (30; 30) in the image, plot the corresponding sinusoid waves in Hough space (rho; theta) and visualize how their intersection point defines the line (what is (m; c) for this line?). Please use MATLAB to plot the curves and report the result in your write-up.*

Answer:

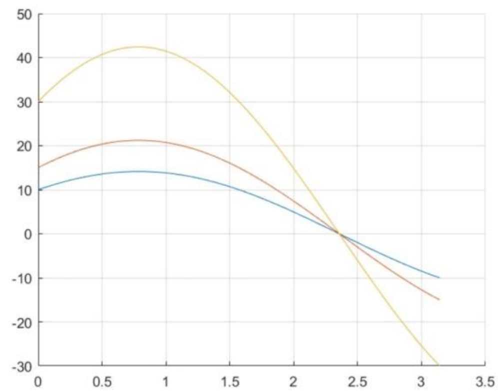
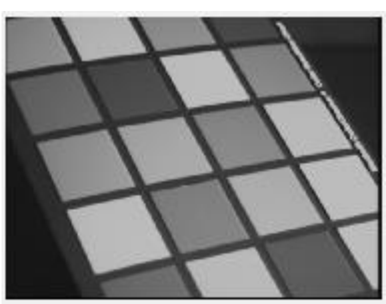


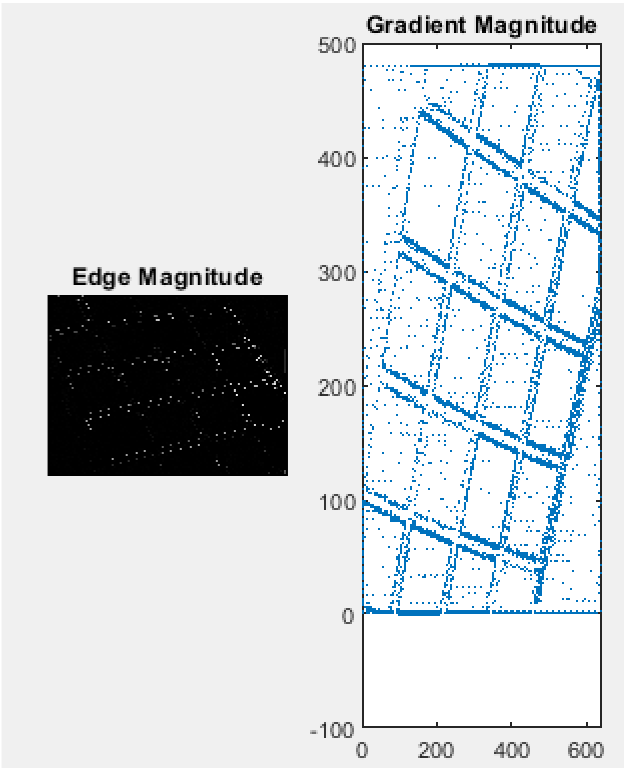
Fig. 1. Hough Transform space

ρ=0 and θ=2.356
m=0.9998 and c=0

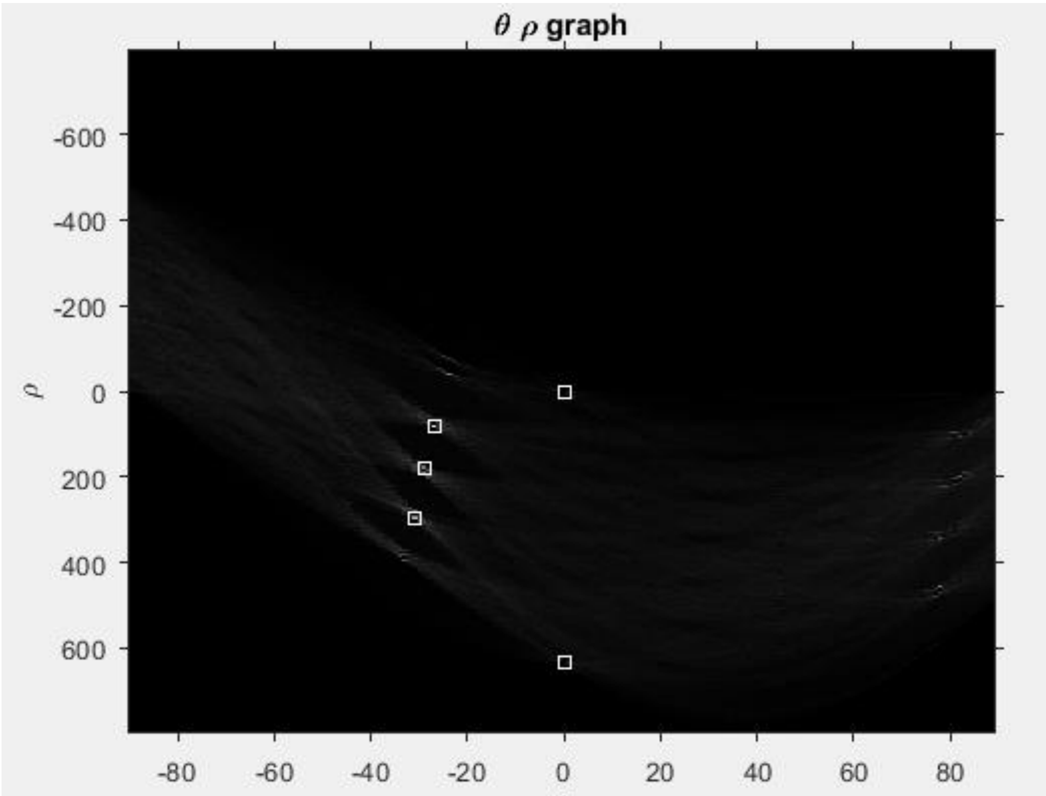
2.1 **Convolution**



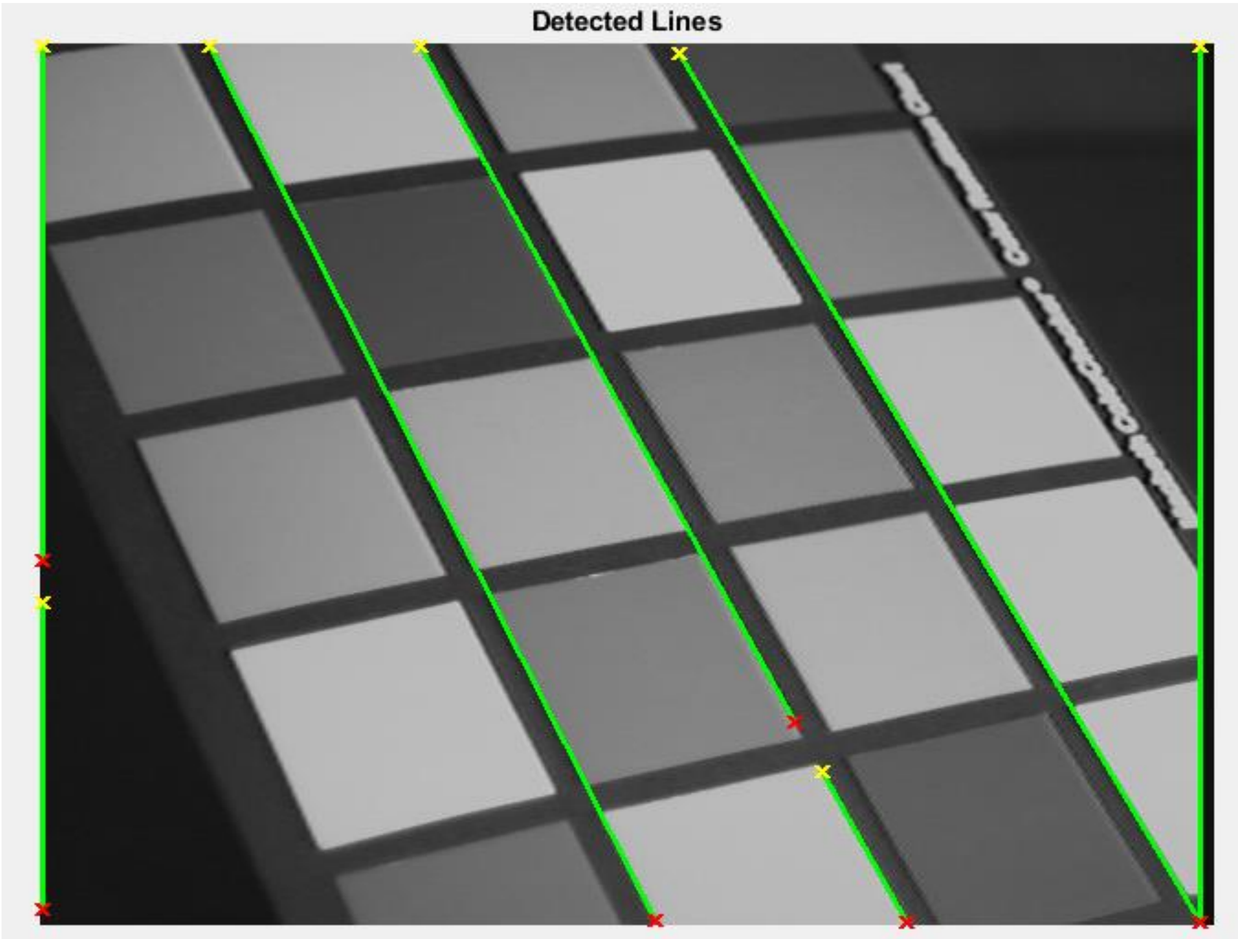
2.2 **Edge Detection**



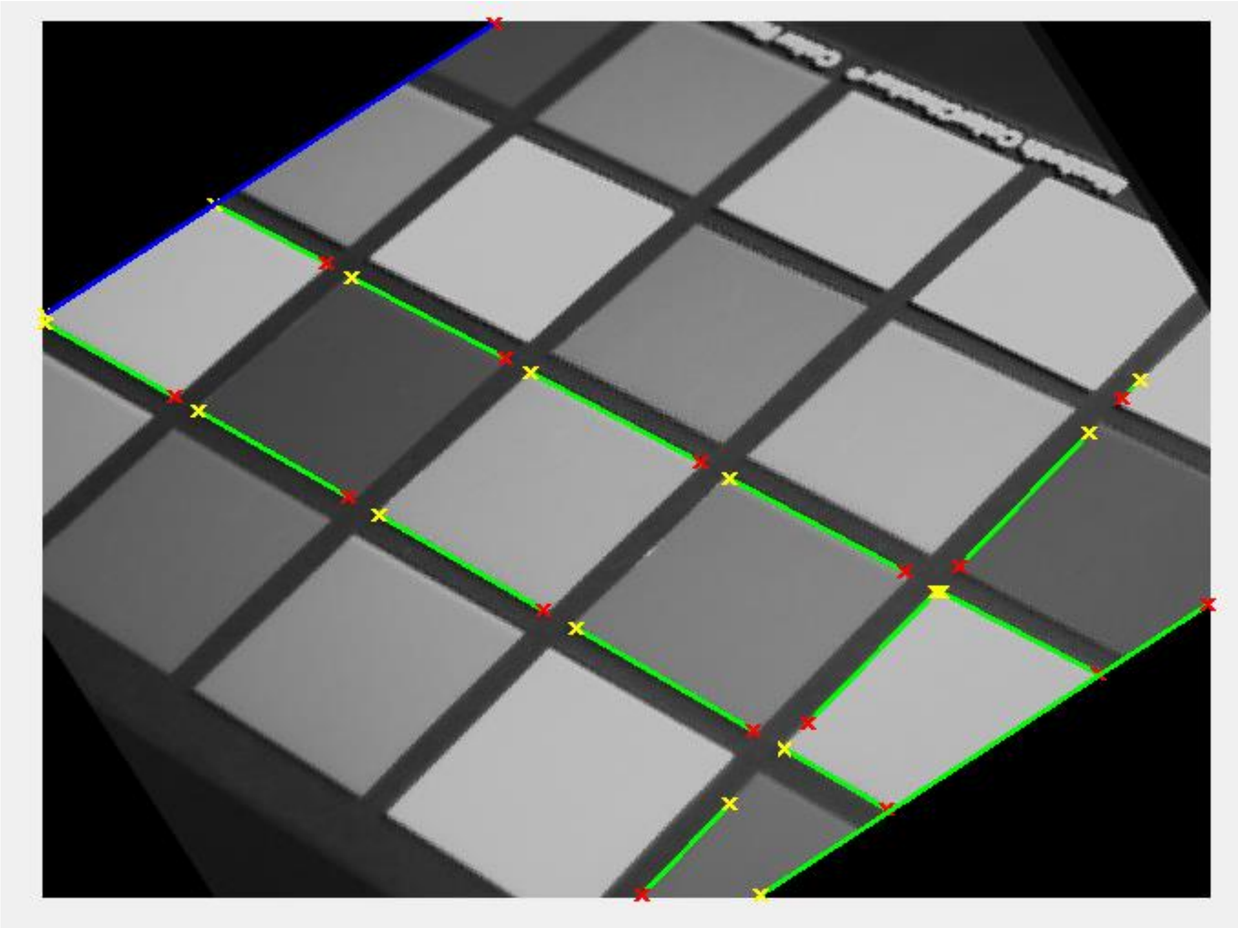
2.3 **The Hough Transform**



2.4 Finding Lines



2.5 **Fitting Line Segments**



3. Experiments

3.1 Use the script included to run your Hough detector on the image set and generate intermediate output images. Include the set of intermediate outputs for one image in your write-up. Did your code work well on all the image with a single set of parameters? How did the optimal set of parameters vary with images? Which step of the algorithm causes the most problems? Did you find any changes you could make to your code or algorithm that improved performance? In your write-up, you should describe how well your code worked on different images, what effect do the parameters have and any improvements you made to your code to make it work better.

Solution:-

The Hough indicator functioned admirably on every one of the pictures with a solitary arrangement of boundaries. The ideal arrangement of boundaries differed with pictures, with the most issues happening in the identification of lines in the pictures with a ton of clamor. I found that the code worked better when the commotion was sifted through before the Hough finder was run. I likewise found that the code worked better when the lines were distinguished in the picture with a higher limit.

Clarification:

The Hough indicator functioned admirably on every one of the pictures with a solitary arrangement of boundaries. The ideal arrangement of boundaries fluctuated with pictures, with the most issues happening in the recognition of lines in the pictures with a ton of clamor. I found that the code worked better when the commotion was sifted through before the Hough locator was run. I likewise found that the code worked better when the lines were distinguished in the picture with a higher limit.

In general, I found that the Hough locator functioned admirably on a large portion of the pictures. The principal issue was with the recognition of lines in the pictures with a ton of clamor. I found that the code worked better when the commotion was sifted through before the Hough locator was run. I likewise found that the code worked better when the lines were recognized in the picture with a higher edge.

A few potential upgrades to the code include:

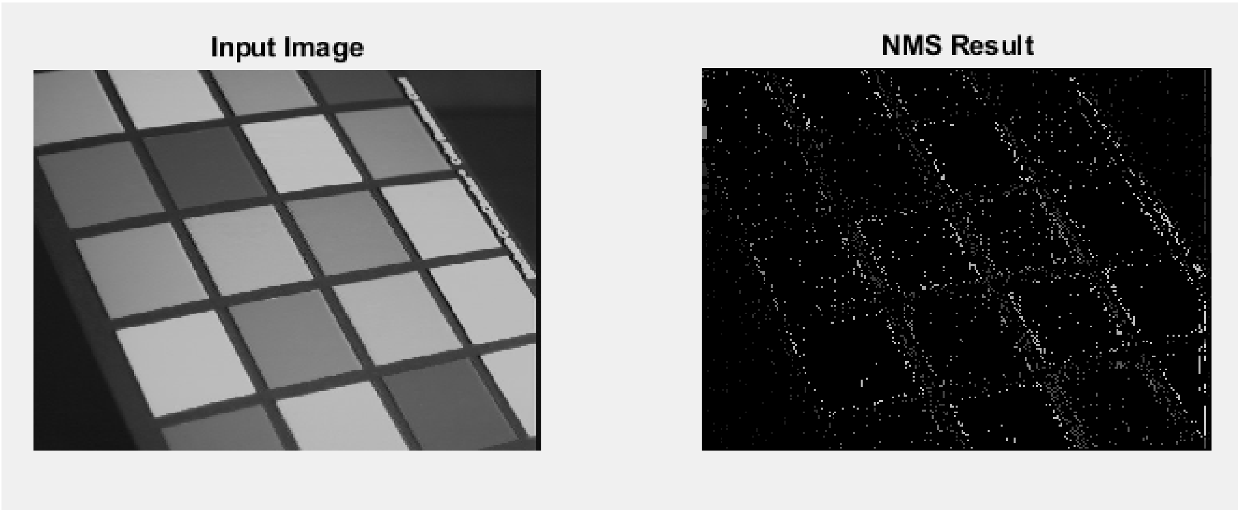
- Utilize a superior sifting technique to eliminate clamor prior to running the Hough locator

- Attempt different Hough indicator boundaries (limit, minLineLength, maxLineGap) on various pictures to check whether improved results can be acquired

- Attempt to identify lines in the picture with a higher edge

- Utilize an alternate technique to recognize lines in the picture (for example Shrewd Edge Indicator)

5. **Non-maximum Suppression**


Input Image
NMS Result