University of New Brunswick

Image Processing Course
(EE 6553)

The Midterm Exam

Professor:
Julian Meng
Electrical and Computer Engineering

Author:
Saeed Kazemi
Student Number:
3713280

December 22, 2020

1 Question 1 (20 points):

Suppose that you form a low-pass spatial filter that averages the four immediate neighbors of a point (x, y) but excludes itself.
a) Find the equivalent filter H(u,v) in the frequency domain.
According to this question, our kernel is similar the below table:

0	$\frac{1}{4}$	0
$\frac{1}{4}$	0	$\frac{1}{4}$
0	$\frac{1}{4}$	0

Now we can write the output function

$$g(x,y) = \frac{1}{4}(f(x,y+1) + f(x+1,y) + f(x-1,y) + f(x,y-1)) \tag{1}$$

$$G(u,v) = \mathcal{F}\{\frac{1}{4}(f(x,y+1) + f(x+1,y) + f(x-1,y) + f(x,y-1))\} \tag{2}$$

$$f(x-x_0, y-y_0) \Longrightarrow F(u,v)e^{-j2\pi(\frac{ux_0}{M}+\frac{vy_0}{N})} \tag{3}$$

$$af(x,y) \Longrightarrow aF(u,v) \tag{4}$$

$$G(u,v) = \frac{1}{4}\mathcal{F}\{f(x,y+1)\} + \mathcal{F}\{f(x+1,y)\} + \mathcal{F}\{f(x-1,y)\} + \mathcal{F}\{f(x,y-1)\} \tag{5}$$

$$G(u,v) = \frac{1}{4}F(u,v)e^{-j2\pi(\frac{-v}{N})} + F(u,v)e^{-j2\pi(\frac{-u}{M})} + F(u,v)e^{-j2\pi(\frac{u}{M})} + F(u,v)e^{-j2\pi(\frac{v}{N})} \tag{6}$$

$$G(u,v) = \frac{1}{4}F(u,v)[e^{-j2\pi(\frac{-v}{N})} + e^{-j2\pi(\frac{-u}{M})} + e^{-j2\pi(\frac{u}{M})} + e^{-j2\pi(\frac{v}{N})}] \tag{7}$$

$$G(u,v) = F(u,v)H(u,v) \tag{8}$$

$$H(u,v) = \frac{1}{4}[e^{-j2\pi(\frac{-v}{N})} + e^{-j2\pi(\frac{-u}{M})} + e^{-j2\pi(\frac{u}{M})} + e^{-j2\pi(\frac{v}{N})}] \tag{9}$$

b) Show that the result is a low-pass filter.
For doing this, we need to show value of filter for different frequencies. I have chosen 9 points.

$\boxed{\text{u=0, v = 0}}$

$$H(0,0) = \frac{1}{4}[e^{-j2\pi(-0/N)} + e^{-j2\pi((-0)/M)} + e^{-j2\pi(0/M)} + e^{-j2\pi(0/N)}] = 1 \tag{10}$$

$\boxed{\text{u=M/2, v=0}}$

$$H(M/2,0) = \frac{1}{4}[e^{-j2\pi(-0/N)} + e^{-j2\pi((-M)/2M)} + e^{-j2\pi(M/2M)} + e^{-j2\pi(0/N)}] = 0 \tag{11}$$

$\boxed{\text{u=M, v = 0}}$

$$H(M,0) = \frac{1}{4}[e^{-j2\pi(-0/N)} + e^{-j2\pi((-M)/M)} + e^{-j2\pi(M/M)} + e^{-j2\pi(0/N)}] = 1 \tag{12}$$

$\boxed{\text{u=0, v = N/2}}$

$$H(0,N/2) = \frac{1}{4}[e^{-j2\pi(-N/2N)} + e^{-j2\pi((-0)/M)} + e^{-j2\pi(0/M)} + e^{-j2\pi(N/2N)}] = 0 \tag{13}$$

$\boxed{\text{u=M/2, v= N/2}}$

$$H(M/2,N/2) = \frac{1}{4}[e^{-j2\pi(-N/2N)} + e^{-j2\pi((-M)/2M)} + e^{-j2\pi(M/2M)} + e^{-j2\pi(N/2N)}] = 0 \tag{14}$$

$\boxed{\text{u=M, v = N/2}}$

$$H(M,N/2) = \frac{1}{4}[e^{-j2\pi(-N/2N)} + e^{-j2\pi((-M)/M)} + e^{-j2\pi(M/M)} + e^{-j2\pi(N/2N)}] = 0 \tag{15}$$

$\boxed{\text{u=0, v = N}}$

$$H(0,N) = \frac{1}{4}[e^{-j2\pi(-N/N)} + e^{-j2\pi((-0)/M)} + e^{-j2\pi(0/M)} + e^{-j2\pi(N/N)}] = 1 \tag{16}$$

$\boxed{\text{u=M/2, v= N}}$

$$H(M/2,N) = \frac{1}{4}[e^{-j2\pi(-N/N)} + e^{-j2\pi((-M)/2M)} + e^{-j2\pi(M/2M)} + e^{-j2\pi(N/N)}] = 0 \tag{17}$$

$\boxed{\text{u=M, v = N}}$

$$H(M,N) = \frac{1}{4}[e^{-j2\pi(-N/N)} + e^{-j2\pi((-M)/M)} + e^{-j2\pi(M/M)} + e^{-j2\pi(N/N)}] = 1 \tag{18}$$

As this table shows, $H(u,v)$ is zero for high frequencies, and passes the low frequencies. The frequency response looks like figure 1.

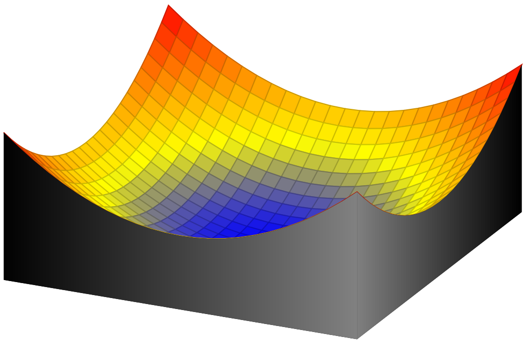


Figure 1: The approximate respond of the filter in frequency domain

	u=0	u=M/2	u=M
v=0	1	0	1
v=N/2	0	0	0
v=N	1	0	1

;

2 Question 2 (20 points):

Consider a circularly symmetric linear spatially invariant (LSI) 2-D imaging system with a 1-D point spread function (PSF) from two subsystems:

$$h_1(x) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp \frac{-x^2}{2\sigma_1^2} \tag{19}$$

$$h_2(x) = \frac{1}{\sqrt{2\pi}\sigma_2} \exp \frac{-x^2}{2\sigma_2^2} \tag{20}$$

a) Find the overall PSF of the overall system.

$$h(x, y) = h_1(x) * h_2(y) \tag{21}$$

$$h(x, y) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{\frac{-x^2}{2\sigma_1^2}} * \frac{1}{\sqrt{2\pi}\sigma_2} e^{\frac{-y^2}{2\sigma_2^2}} \tag{22}$$

$$h(x, y) = \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{x^2}{2\sigma_1^2} + \frac{-y^2}{2\sigma_2^2}} \tag{23}$$

if $\sigma_1 = \sigma_2 = \sigma$

$$h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{24}$$

otherwise

$$h(x, y) = \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{(\sigma_2^2 x^2 + \sigma_1^2 y^2)}{2\sigma_1^2\sigma_2^2}} \tag{25}$$

b) Determine the FWHM of the overall system. FWHM can be determined as the distance between the curve points at the peak half maximum level. Figure 2 shows the FWHM.

$$h_1(x) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp \frac{-x^2}{2\sigma_1^2} \tag{26}$$

if $x = 0$

$$h_1(0) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp 0 = \frac{1}{\sqrt{2\pi}\sigma_1} \Longrightarrow \textit{max value of } h_1 \tag{27}$$

half of max value is

$$\frac{1}{2\sqrt{2\pi}\sigma_1} \Longrightarrow half\ max\ value\ of\ h_1 \tag{28}$$

$$\frac{1}{2\sqrt{2\pi}\sigma_1} = \frac{1}{\sqrt{2\pi}\sigma_1} \exp\frac{-x^2}{2\sigma_1^2} \tag{29}$$

By logging on both side

$$\ln\frac{1}{2} = \frac{-x^2}{2\sigma_1^2} \tag{30}$$

$$-2\sigma_1^2 \ln 2 = -x^2 \tag{31}$$

By using square root on both side

$$x = \pm\sqrt{2\sigma_1^2 \ln 2} = \pm\sigma_1\sqrt{2\ln 2} = \pm 1.177\sigma_1 \tag{32}$$

$$x_1 = +1.177\sigma_1 \quad x_2 = -1.177\sigma_1 \tag{33}$$

$$FWHM_{h_1} = x_1 - x_2 = +1.177\sigma_1 - (-1.177\sigma_1) = 2.35\ \sigma_1 \tag{34}$$

similarly

$$FWHM_{h_2} = 2.35\ \sigma_2 \tag{35}$$

Since the system is LTI, FWHM of the overall system is

$$FWHM_h = FWHM_{h_1} * FWHM_{h_2} = 2.35\ \sigma_2 * 2.35\ \sigma_2 = 5.53\ \sigma_2\sigma_1 \tag{36}$$

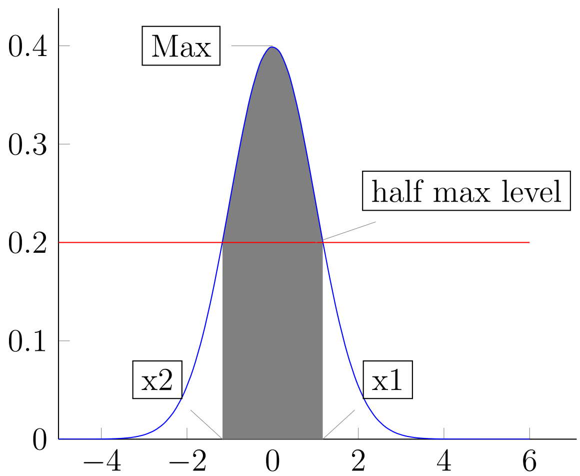


Figure 2: the FWHM

3 Question 3 (25 points):

Let the original image, f(x,y), and distorted image, g(x,y), be zero mean random processes where

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y) \tag{37}$$

Prove that the minimum mean square error (Wiener) filter solution to

$$e^2 = E\{(f(x,y) - \hat{f}(x,y))^2\} \tag{38}$$

in the frequency domain is

$$H_{Wiener}(u,v) = [\frac{1}{H(u,v)}[\frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{S_\eta(u,v)}{S_f(u,v)}}]] \tag{39}$$

where

$$\hat{F}(u,v) = H_{Wiener}(u,v)\ G(u,v) \tag{40}$$

Hint: Minimization of e^2 requires $(f(x,y) - \hat{f}(x,y))$ and g(x,y) to be orthogonal.

$$e^2 = E\{(f(x,y) - \hat{f}(x,y))^2\} \tag{41}$$

So we need minimization of below equation:

$$e^2 = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} [f(x,y) - \hat{f}(x,y)]^2 dxdy \tag{42}$$

Based on Parseval's Theorem

$$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |f(x,y) - \hat{f}(x,y)|^2 dxdy = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} |F(u,v) - \hat{F}(u,v)|^2 dudv \tag{43}$$

also we know that

$$\hat{F}(u,v) = H_W(u,v)G(u,v) = H_W(u,v)H(u,v)F(u,v) + H_W(u,v)N(u,v) \tag{44}$$

$$F - \hat{F}(u,v) = (1 - H_W(u,v)H(u,v))F(u,v) - H_W(u,v)N(u,v) \tag{45}$$

so

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}|(1-H_W(u,v)H(u,v))F(u,v)-H_W(u,v)N(u,v)|^2dudv = \tag{46}$$

since f(x,y) and n(x,y) are uncorrelated and $E\{n(x,y)\}=0$

$$E\{f(x,y)n(x',y')\} = E\{f(x,y)\}E\{n(x',y')\} = 0 \tag{47}$$

then

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}|(1-H_W(u,v)H(u,v))F(u,v)|^2+|H_W(u,v)N(u,v)|^2dudv = \tag{48}$$

now to minimize integral we need to minimize integrand for all (u,v)

$$\frac{\partial}{\partial A}(|A|^2) = \frac{\partial}{\partial A}(A^*A) = A^*\frac{\partial}{\partial A}(A) + A\frac{\partial}{\partial A}(A^*) \tag{49}$$

So

$$\frac{\partial}{\partial H_W}[|(1-H_W(u,v)H(u,v))F(u,v)|^2+|H_W(u,v)N(u,v)|^2] = 0 \tag{50}$$

$$-H(u,v)F(u,v)((1-H_W^*(u,v)H^*(u,v))F^*(u,v))+(H_W^*(u,v)N^*(u,v))(N(u,v)) = 0 \tag{51}$$

$$(H_W^*(u,v)H^*(u,v)-1)H(u,v)|F(u,v)|^2+H_W^*(u,v)|N(u,v)|^2 = 0 \tag{52}$$

$$H_W^*(u,v) = \frac{H(u,v)|F(u,v)|^2}{|H(u,v)|^2|F(u,v)|^2+|N(u,v)|^2} \tag{53}$$

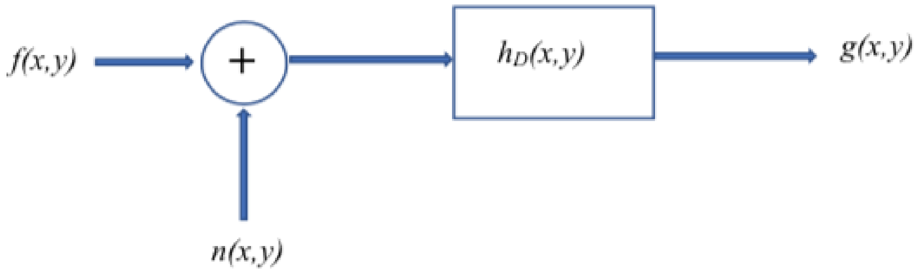
$$H_W(u,v) = \frac{H(u,v)^*|F(u,v)|^2}{|H(u,v)|^2|F(u,v)|^2+|N(u,v)|^2} \tag{54}$$

$$H_W(u,v) = \frac{H(u,v)^*}{|H(u,v)|^2+\frac{|N(u,v)|^2}{|F(u,v)|^2}} \tag{55}$$

$$H_W(u,v) = [\frac{1}{H(u,v)}[\frac{|H(u,v)|^2}{|H(u,v)|^2+\frac{S_\eta(u,v)}{S_f(u,v)}}]] \tag{56}$$

4 Question 4 (25 points):

Consider the following imaging system (please note the placement of the additive noise, this differs from the "classical" position of after the LSI block):



where an input wave image, $f(x,y) = 10sin(2\pi u_0 x)$, where $u_0 = 0.6mm^{-1}$ is the spatial frequency, is corrupted with AWGN, $n(x,y)$, with a $PSD_\eta = 1Wmm$. The corrupted image is the input to a degradation function with a transfer function of $H_D(u,v) = 1 - \frac{|u|}{B}$ for $|u| \leq B$ and $H_D(u,v) = 1$ for $|v| \leq B$ where the cut-off frequency is $B = 1.2mm^{-1}$, to form an image $g(x,y)$.

a) What is the mean and variance of the noise in the output, $g(x,y)$?
For this part, because the system is a linear, we can consider f(x, y) = 0 to evaluate noise at the output. So we have

$$g(x,y) = h(x,y) * [f(x,y) + n(x,y)] \tag{57}$$

if f(x, y) = 0 then

$$g(x,y) = h(x,y) * n(x,y) \tag{58}$$

$$G(u,v) = H(u,v)N(u,v) \tag{59}$$

$$\mu_g = E\{g(x,y)\} = E\{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(k,l)n(x-k,y-l)dkdl\} \tag{60}$$

$$\mu_g = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(k,l)E\{n(x-k,y-l)\}dkdl \tag{61}$$

$$\mu_g = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(k,l)\mu_n dkdl \tag{62}$$

$$\mu_g = \mu_n \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(k,l)dkdl \tag{63}$$

we know that

$$H(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(k,l)e^{-2j\pi(ux+vy)}dxdy \tag{64}$$

therefore

$$\mu_g = \mu_n H(0,0) \tag{65}$$

$$\boxed{\mu_g = \mu_n} \tag{66}$$

$$var(g) = C_g(t) = R_g(t) - \mu_g^2(t) \tag{67}$$

$$R_g(t) = E\{g(x,y)g(x,y)\} = E\{g^2(x,y)\} \tag{68}$$

$$R_g(t) = E\{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^2(k,l)n^2(x-k,y-l)dkdl\} \tag{69}$$

$$R_g(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^2(k,l)E\{n^2(x-k,y-l)\}dkdl \tag{70}$$

$$R_g(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^2(k,l)R_n(t)dkdl \tag{71}$$

$$R_g(t) = R_n(t)\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^2(k,l)dkdl \tag{72}$$

According to Parseval's theorem and real transfer function

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |h(x,y)|^2dxdy = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |H(u,v)|^2dudv \tag{73}$$

$$R_g(t) = R_n(t)\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |H(u,v)|^2dudv \tag{74}$$

$$\boxed{Var(g) = AR_n(t) - \mu_n^2} \tag{75}$$

where A is

$$A = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |H(u,v)|^2dudv \tag{76}$$

b) What is the SNR for the input and output image of the LSI system?

$$SNR = 10\log_{10}\frac{S}{N} \tag{77}$$

where S and N are the power of signal and noise, respectively. In the input side, Signal power for sine signal is

$$S = \frac{10^2}{2} = 50 \tag{78}$$

also

$$N = \int_{-\infty}^{\infty} S_n(f)df = \int_{-\infty}^{\infty} 1df = \infty \tag{79}$$

So

$$SNR = 10\log_{10}\frac{50}{\infty} \approx -\infty \ \ dB \tag{80}$$

In the output side, we know

$$S_g(f) = S_f(f)|H_D(f)|^2 \tag{81}$$

for noise, we have

$$S_{g_n}(f) = S_n(f)|H_D(f)|^2 \tag{82}$$

$$S_{g_n}(f) = 1 * |H_D(f)|^2 \tag{83}$$

$$N = \int_{-B}^{B} S_{g_n}(f)df \tag{84}$$

by helping Matlab, I solved the integral, so

$$N = \frac{2}{3}B = \frac{2 * 1.2}{3} = 0.8 \tag{85}$$

for signal, we did the same process

$$S_{g_f}(f) = S_f(f)|H_D(f)|^2 \tag{86}$$

$$S_f(f) = \frac{10^2}{4}[\delta(f - f_0) + \delta(f + f_0)] \tag{87}$$

$$H_D(0.6) = 1 - \frac{0.6}{1.2} = 0.5 \tag{88}$$

$$|H_D(0.6)|^2 = 0.5^2 = 0.25 \tag{89}$$

$$S = \int_{-\infty}^{\infty} S_{g_f}(f)df = \int_{-\infty}^{\infty} 0.25 S_f(f)df \tag{90}$$

$$S = 0.25 * 2 * \frac{10^2}{4} = 12.5 \tag{91}$$

So

$$\boxed{SNR = 10\log_{10}\tfrac{12.5}{0.8} \approx 12 \ \ dB} \tag{92}$$

c) Propose an image restoration system applied to $g(x, y)$ to improve SNR and re-calculate results for part b).

Personally, I think the best system for improving SNR is a band pass filter. This filter will decline the power of Noise at the output. we call it H_R.

$$H_R(f) = \begin{cases} 1, & \text{if } f = 0.6\ mm^{-1}. \\ 0, & \text{otherwise.} \end{cases} \tag{93}$$

In the output of band pass filter, we have

$$S_g(f) = S_f(f)|H_D(f)H_R(f)|^2 \tag{94}$$

for noise, we have

$$S_{g_n}(f) = S_n(f)|H_D(f)H_R(f)|^2 \tag{95}$$

$$S_{g_n}(f) = 1 * |H_D(f)H_R(f)|^2 \tag{96}$$

$$N = \int_{-B}^{B} S_{g_n}(f)df \tag{97}$$

by helping Matlab, I solved the integral, so

$$N = 0.25 \tag{98}$$

for signal, we did the same process

$$S_{g_f}(f) = S_f(f)|H_D(f)H_R(f)|^2 \tag{99}$$

$$S_f(f) = \frac{10^2}{4}[\delta(f - f_0) + \delta(f + f_0)] \tag{100}$$

$$H_D(0.6) = 1 - \frac{0.6}{1.2} = 0.5 \tag{101}$$

$$|H_D(0.6)|^2 = 0.5^2 = 0.25 \tag{102}$$

$$S = \int_{-\infty}^{\infty} S_{g_f}(f)df = \int_{-\infty}^{\infty} 0.25 S_f(f)df \tag{103}$$

$$S = 0.25 * 2 * \frac{10^2}{4} = 12.5 \tag{104}$$

So

$$\boxed{SNR = 10\log_{10}\tfrac{12.5}{0.25} \approx 17\ \ dB} \tag{105}$$

5 Question 5 (35 points):

This project explores a method to detect straight lines in an image from the collection of isolated edge-pixels generated by an edge detector algorithm. A straight line in a 2-D x-y plane can be characterized by its distance from the origin, $d \in (0, +\infty)$ and the angle $\theta \in (0, 2\pi)$, between the normal to the line and the x–axis (usually in a counter-clockwise sense). That is,

$$xcos(\theta) + ysin(\theta) = d \tag{106}$$

Reciprocally, any (θ, d) uniquely characterize a set of x's and y's that define a straight line given by the above. Given a single pixel, $M_0 = (x_0, y_0)$, we can characterize in the (θ, d) plane the set of all lines that pass through M_0 . This set is comprised by the graph of the sinusoidal function:

$$d = x_0\ cos(\theta) + y_0\ sin(\theta), \theta \in (0, \pi) \tag{107}$$

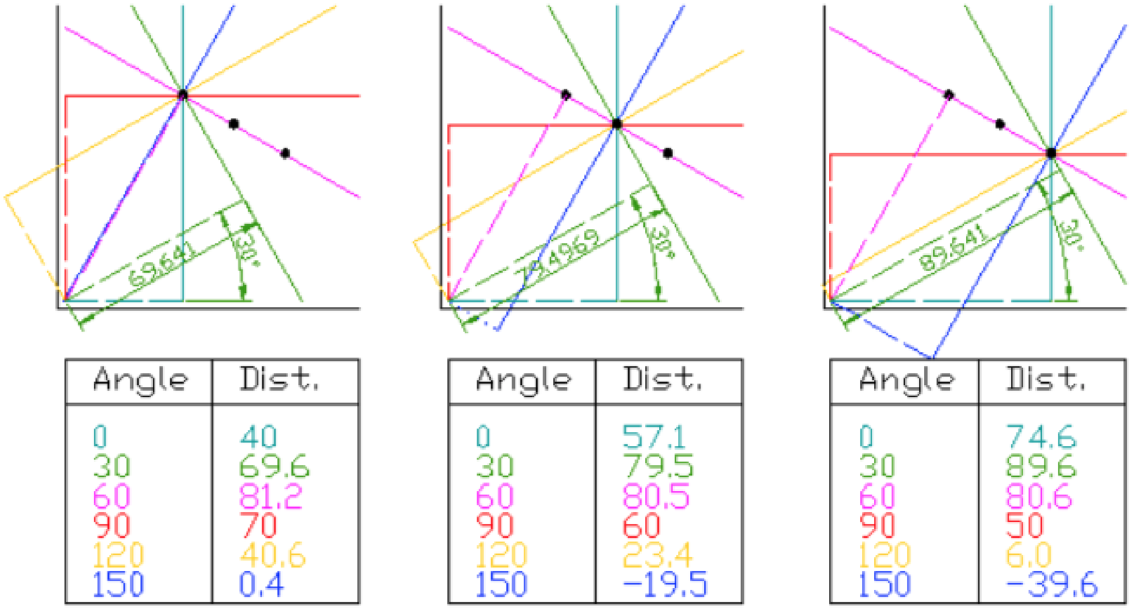
The figure below shows three separate pixels at (x, y) points and the corresponding sinusoidal functions in the (θ, d) plane. θ has been discretized to 0, 30, 60, 90, 120 and 150 degrees. The values in the table represent to output of the Hough Transform in the (θ, d) plane.

If N pixels, $M_i = (x_i, y_i), i = 0, 1, 2, ..., N - 1$, are aligned along a straight line, then the N sinusoidal functions:

$$d = x_i cos(\theta) + y_i sin(\theta), \theta \in (0, 2\pi)$$

will all intersect at a single point (θ^*, d^*) that defines a straight line that passes through M_i . This is the principal of detecting lines using the Hough Transform – a line of points or pixels converge to a single point in the (θ, d) plane, as shown below:

In the above example, the intersection point is $(\theta^* = 60°, d^* = 81)$ with an intensity of 3 in the Hough domain and this represents the purple line in the (x, y) domain. Theoretically, the intersection point should be unique, however, in practice, the effect of noise combined with the discretization of θ and d will create some uncertainty of the location of the intersection, so one must use a threshold technique to determine if a pixel is part of the line or not. For this task,



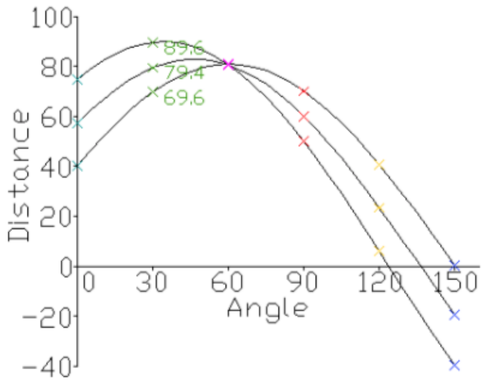
Source: Wikipedia

a) Develop an algorithm that implements the Hough Transform.
For this part, I used the below algorithm for implementing, and based on this algorithm, I wrote a code.

1. First change input image to binary image
2. Then scan the binary image to find one values in the binary matrix (by using two for loops) and store location in for $(x_i, y_i), i = 0, 1, 2, ...$
3. After that for each stored pixel $(x_i, y_i), i = 0, 1, 2, ...$ finds d values for each value in the theta vector based on the below equation.

$$for\ (x_i, y_i), i = 0, 1, 2, ...$$
$$\vec{\Theta} = [\theta_1, \theta_2, \dots, \theta_N] \in (0, 2\pi)$$
$$d = x_i cos(\theta) + y_i sin(\theta)$$
$$\vec{d} = [d_1, d_2, \dots, d_N]$$

4. Then make a matrix for different values in the θ and d called H



Source: Wikipedia

5. Then add an unit to H matrix in location (d, θ). In other words, $H(d, \theta) = H(d, \theta) + 1$ for
6. Then return H as a Hough Transform
7. the max location of H matrix shows the θ and d of line in the image

b) Verify that applying this algorithm to a set of pixels in a straight line yields a bright point in the Hough Transform domain.

For this question, we set four points with one. these points were in a straight line (see figure 3). Due to we had four points, the output image had four curves. Figure 4 illustrates these curves. The intersection of this plot gives information about the line (The θ and d).

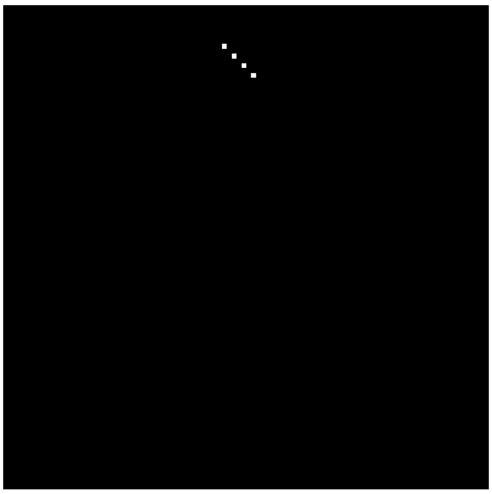


Figure 3: the input image for testing algorithm

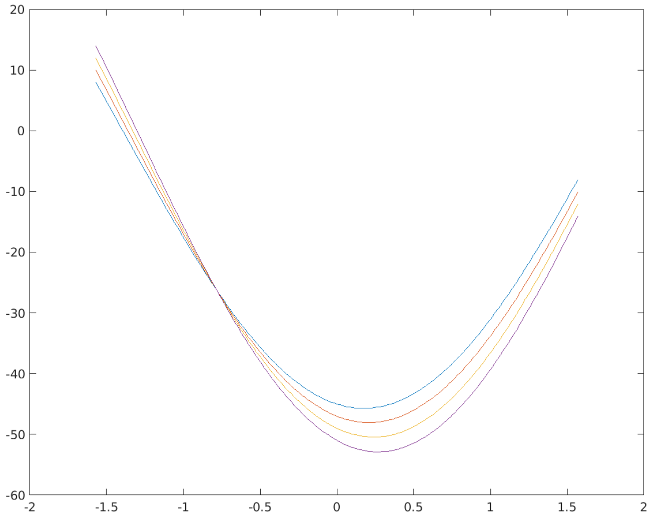


Figure 4: the plot of different d and θ.

```
clearvars;
clc
close all


BW = zeros(100,100);
BW(09,46)=1;
BW(11,48)=1;
BW(13,50)=1;
BW(15,52)=1;
figure;imshow(BW)



theta = ((-90:90)./180) .* pi;
D = sqrt(size(BW,1).^2 + size(BW,2).^2);
H = zeros(ceil(2.*D),size(theta,2));


k=1;
for i=1:size(BW,1)
    for j=1:size(BW,2)
        if BW(i,j)==1
            p(k,1)=i;
            p(k,2)=j;
            k=k+1;
        end
    end
```

```
end

y = p(:,1)-1;
x = p(:,2)-1;



R = cell(1,size(x,1));


for i = 1: size(x,1)
    R{i} = x(i).*cos(theta) + y(i).*sin(theta);
    plot(theta,-R{i})
    hold on
end


for i = 1:size(x,1)
    R{i} = R{i} + D;
    R{i} = floor(R{i}) + 1;
    for j = 1:size(R{i},2)
        H(R{i}(j),j) = H(R{i}(j),j) + 1;
    end
end
figure;imshow(H)
```

c) Describe an algorithm to detect all straight lines in an image.
In this algorithm, each point in the image that has a value more than zero is selected. Then for these points all of potential lines are calculated. At the end of this process, we find lines that happened frequently by finding the max values, and return the θ and d as a location line in image.

d) Implement and test this algorithm on an image of your choice. Make sure to use an image with lots of linear features.
As can be seen, there are four peak in figure 6. the location of these peak were shown in figure 7. Each peak represents a line. For example, (-1.57, 75) indicates that we have a line with $\theta = -\frac{\pi}{2}$ and d = 74.

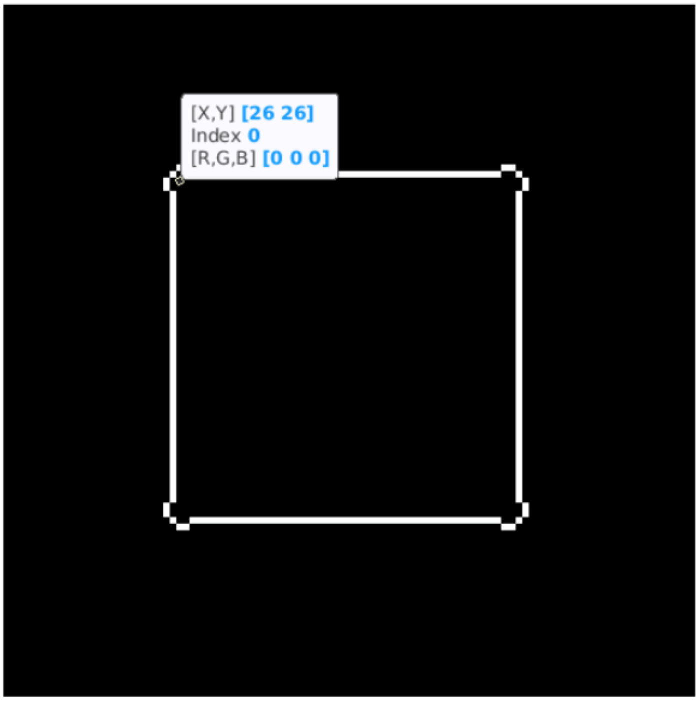


Figure 5: input image

Figure 6: the image of H matrix

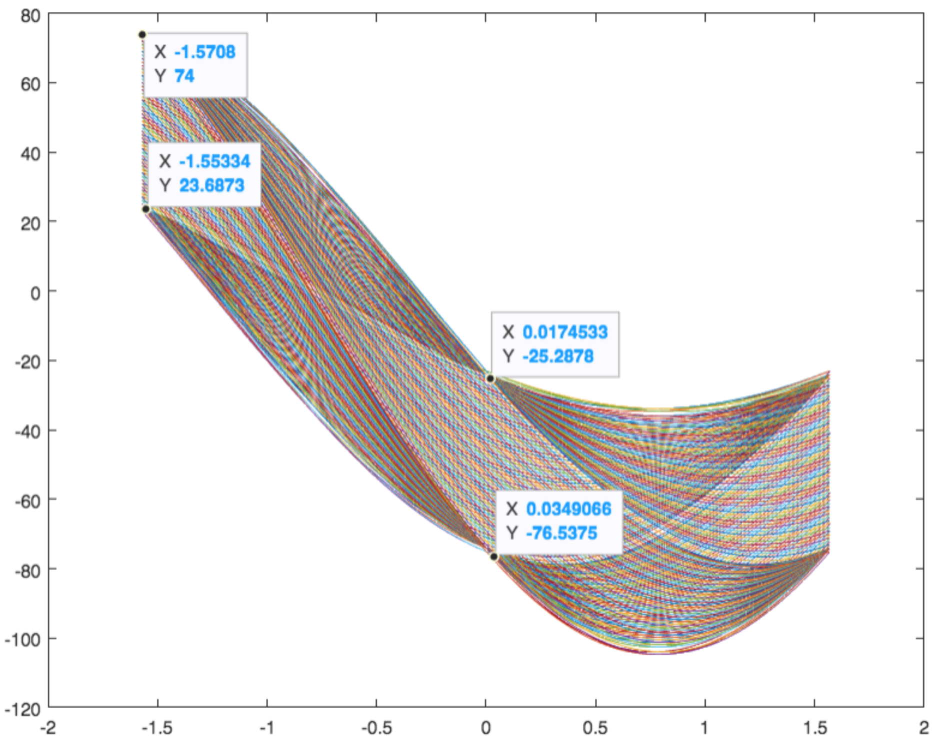


Figure 7: the plot of different d and θ.