2020-11120 GENE CHUNG

#1 $10\times 8 \quad * \quad m\times n = 8\times 8$

$\therefore m\times n = 3\times 1$

→ $\begin{matrix}0\\0\\0\end{matrix} * \begin{matrix}a\\b\\c\end{matrix} = 0 \qquad \begin{matrix}0\\0\\1\end{matrix} * \begin{matrix}a\\b\\c\end{matrix} = -1$

$\begin{matrix}0\\1\\1\end{matrix} * \begin{matrix}a\\b\\c\end{matrix} = 1 \qquad \begin{matrix}1\\1\\1\end{matrix} * \begin{matrix}a\\b\\c\end{matrix} = 0$

→ $c=-1 \quad b=2 \quad a=-1 \qquad \therefore ? = \begin{matrix}-1\\2\\-1\end{matrix}$

and this are similar to Δ^2 operator which is $Y_{i+2} - 2Y_{i+1} + X_i$ but having different sign.

This operator detects a point changing rapidly in column direction (Edge detection)

#2

2-1. When does the zero-crossing occurs?

The answer will be point where laplacian's sign changes. It is usually when intensity of the image changes rapidly. where? Edges!
So usually Edges corespond to zero-crossings.

2-2. bigger sigma makes gausscan-filter bigger which makes more of blur and reduces more zero-crossing points and make less points where h(x,y) becomes zero since large σ detects large scale edges & small σ detects fine features

#3 3-1. $\rho = x\sin\theta + y\cos\theta$ for fixed x, y

$\rightarrow \rho = x\sin\theta + y\cos\theta = \sqrt{x^2+y^2}\cos\alpha\sin\theta + \sin\alpha\cos\theta$

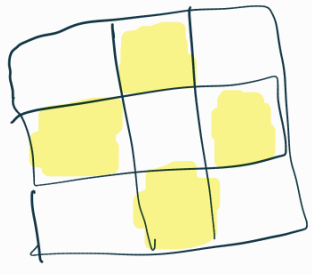
$(\alpha = \arctan(\frac{y}{x}))$

$\Rightarrow \therefore \rho = \sqrt{x^2+y^2}\sin(\alpha+\theta)$

$\therefore$ phase shifted $-\arctan(\frac{y}{x})$ & amplitude be $\sqrt{x^2+y^2}$ //

3-2. period/frequency doesn't varies with the point (X, Y) & stay constant as 2π in $\theta \rightarrow$ AXIS //

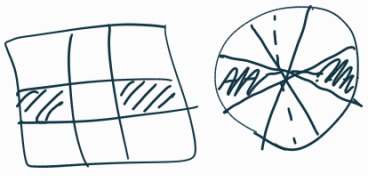
#4. Method I used while Implement non maximal Suppression

→ 

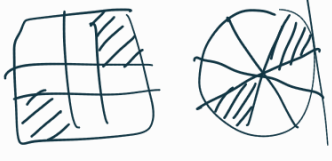
→ there are total 8 surrandy we can use only 4 neighboring pixels, but to make more accurate non-maximul Suppression, I used all 8 surroundings,

First, I check Gradient Direction derived by $\theta = \arctan\left(\frac{S_Y}{S_X}\right)$ and selected two neighborcy pixels by such standard below: $-\frac{\pi}{2} < \theta < \frac{\pi}{2}$

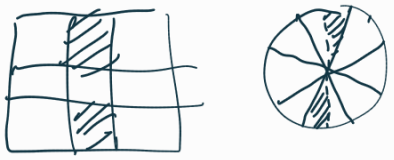
(i) $-\frac{\pi}{8} < \theta \leq \frac{\pi}{8}$



(ii) $\frac{\pi}{8} < \theta \leq \frac{3\pi}{8}$



(iii) $\frac{3\pi}{8} < \theta \leq \frac{\pi}{2}$ or $-\frac{3\pi}{8} > \theta > -\frac{\pi}{2}$



(iv) $-\frac{3}{8}\pi < \theta < -\frac{\pi}{8}$

→ Double Thresholding

At weak edge I detected the edge by comparing 4 neighbourcy pixels and with highthreshold and If all the 4 values were less then highthreshold I made the value to 0

By changing Parameter we could obtain 4 different pictures fig1.c has lots of noice since I use low sigma 2 So I changed Sigma to 3 and obtained Frg a,b,d → Frg a used

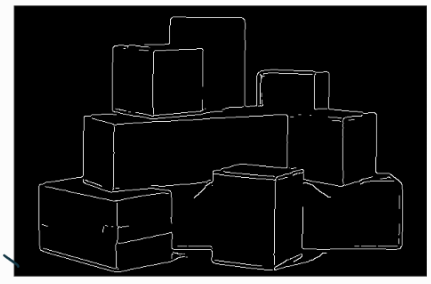


Fig 1.a

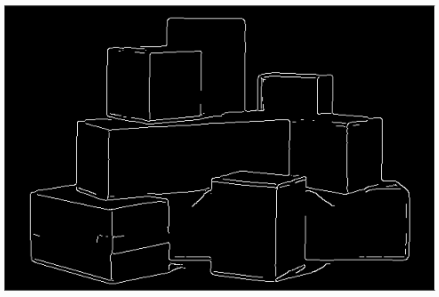


Fig 1.b

high threshold 0.14
low Threshold 0.04

b used high 0.14
low 0.02

d used high 0.05
low 0.02

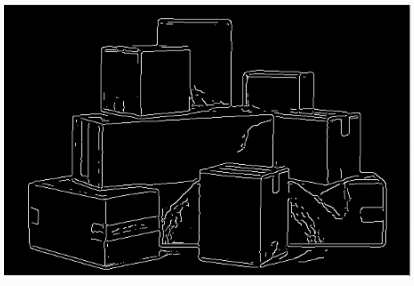


Fig 1.c

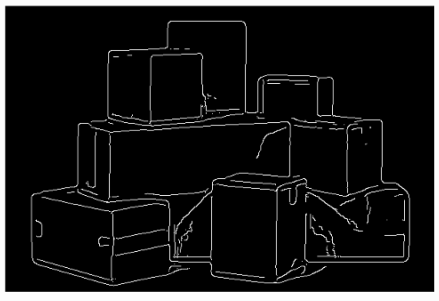


Fig 1.d

There wee no dramatic change by low threshold but changing gausskernel helped reducing noise.

4.3 Hough transform.

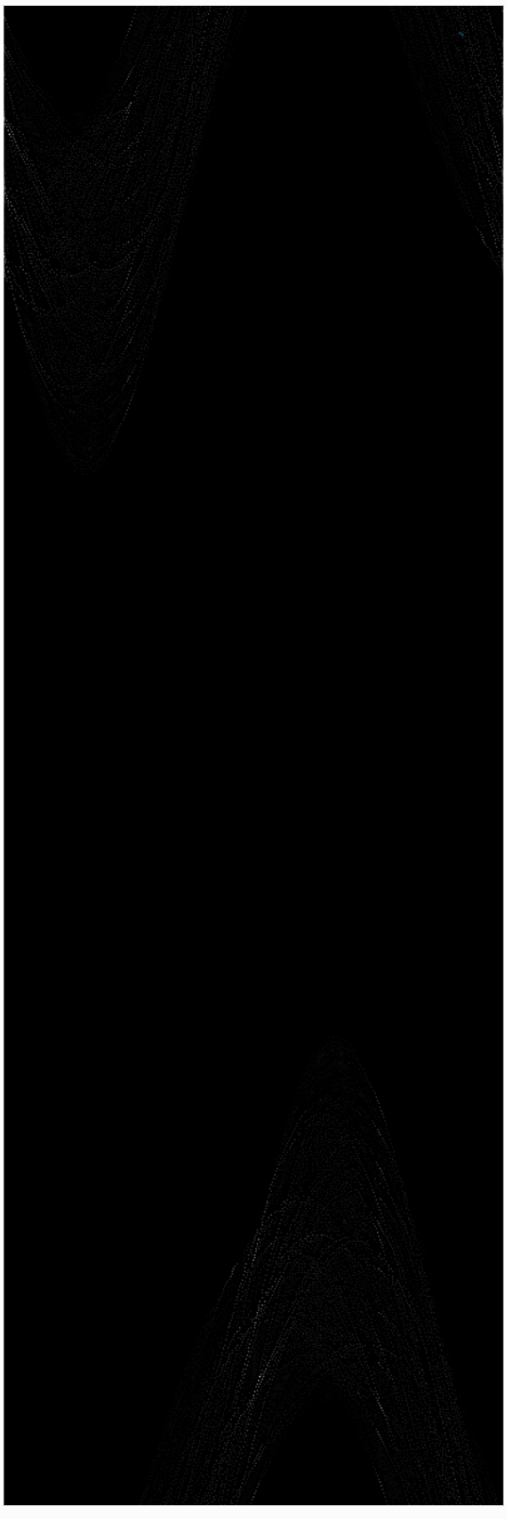


Fig 2.9

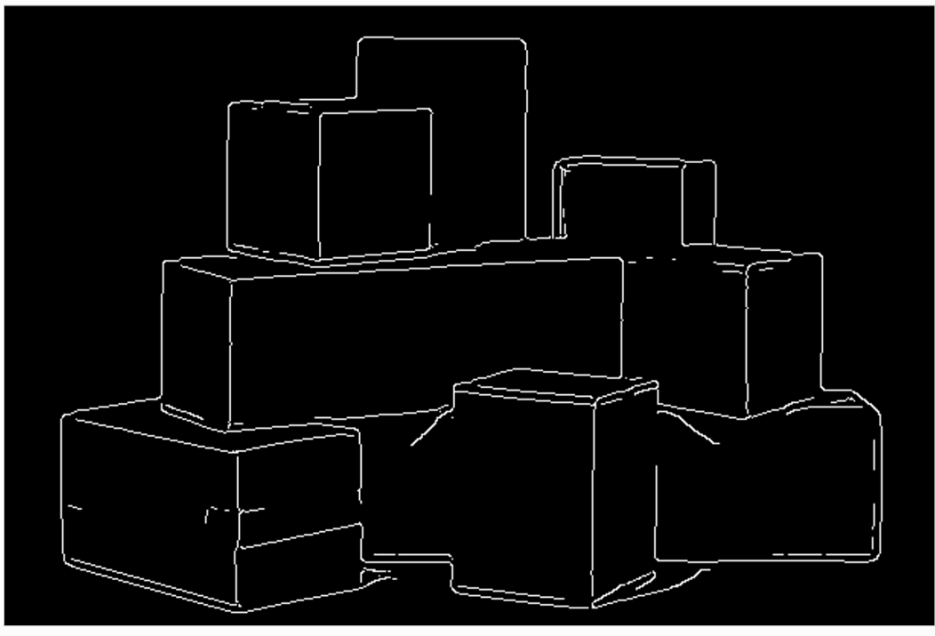


Fig 2.b

Fig 2.a shows hough trasform stored functions on coordinate (ρ, θ) & Fig 2.b shows Im image. I used parameter of rhoRes=1.8

thetaRes=math.pi/200

& rhoMaxNum=1200 which indicates maximum # of coordinate 'rho'.

4-4.

r_p & r_θ are also needed here since returning lRho, lTheta should be the real 'rho' & 'theta' value, not the int value decided by resolution. So we need each resolution values.

I used all 8 neighbourg pixels for

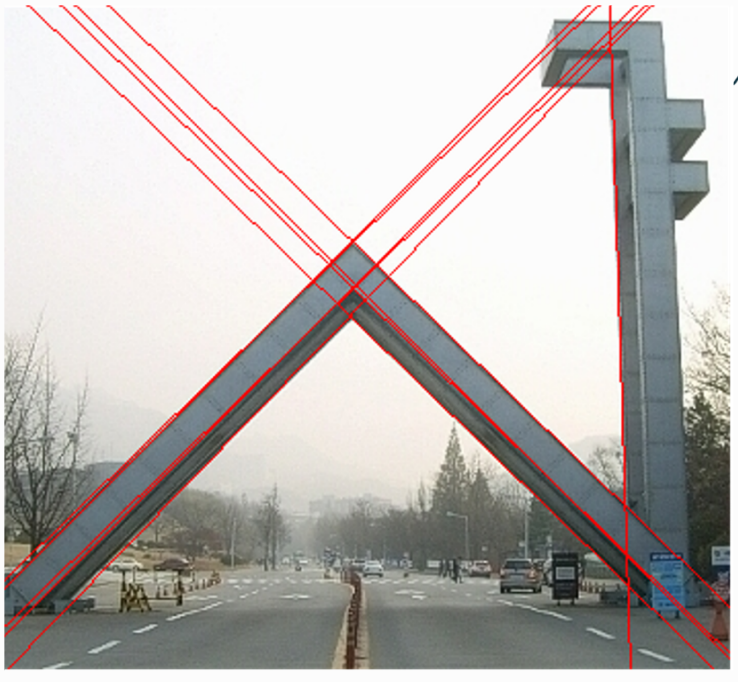


Fig 3. a

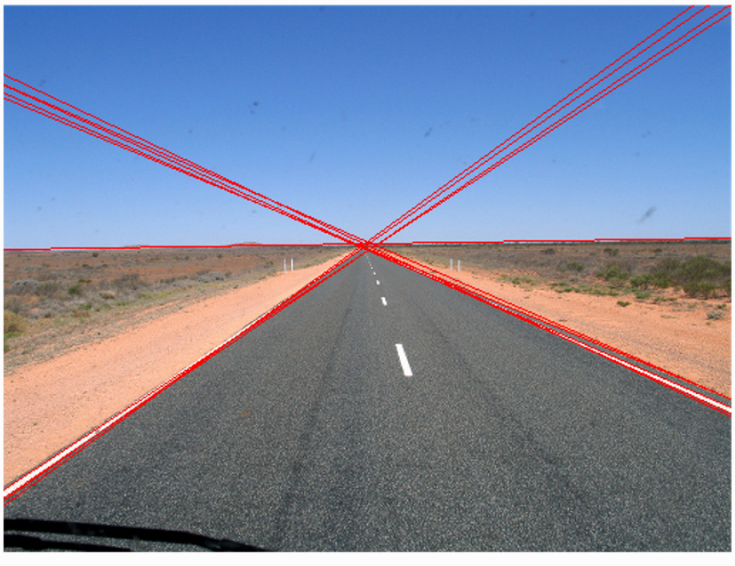


Fig 3. b

Fig 3.a & 3.b is both hough line with orgonal picture

#5

The main algorith used to detect the line segments was by detecting the point on the line and see if its out of the edge or It's the edge. I used algorithm of offset and seek of pexels around in boundary of offset has a point H[i][5]=1 Then I used to get segments line

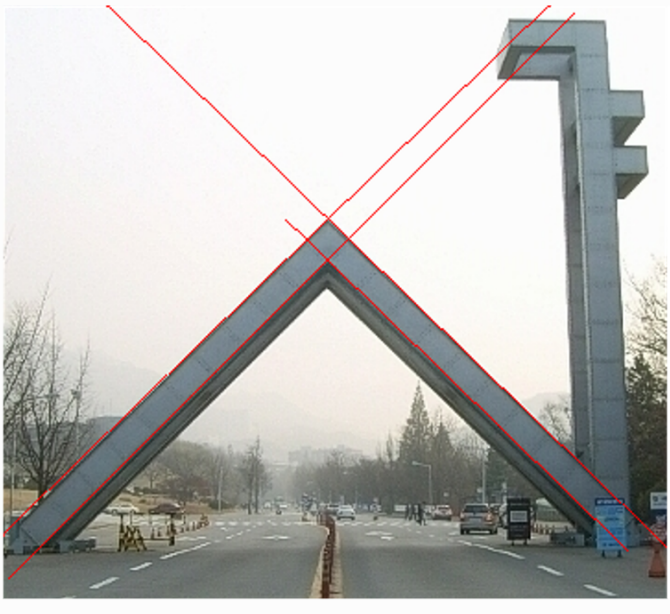


Fig 4.a

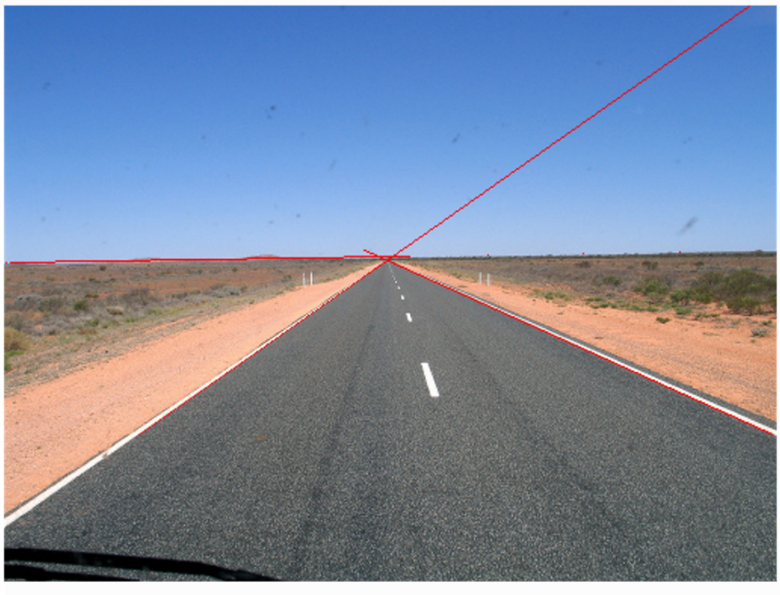


Fig 4.b