FINALS

Date 29th November, 2021

(Fourier trans.)

$f(x,y) = \begin{bmatrix} 5 & 3 \\ 1 & 2 \end{bmatrix} \xrightarrow{F(u,v)} F(u,v) = \begin{bmatrix} 11 & 1 \\ 5 & 3 \end{bmatrix}$ ← freq. coefficients

↓ apply freq. filter (high pass / low pass)

$g(x,y) \xleftarrow{g(x,y) = F^{-1}(u,v)}$ (inverse trans.)

$F'(u,v) = F(u,v) * w(s,t)$ (convolution)

• why we do transformation? Some problems are hard to remove in spatial domain than in freq. domain e.g. periodic noise

• if we inverse $\begin{bmatrix} 11 & 1 \\ 5 & 3 \end{bmatrix}$ we will get original image

↑ we have to apply filter on freq. coefficients and not original image because time complexity.

Inverse Transformation:

$$I(r,c) = k' \sum_{r=0}^{1} \sum_{c=0}^{1} T(u,v) \cdot B(r,c,u,v)$$ let $k' = 1$

$I(0,0) = 11 \times 1 + 1 \times 1 + 5 \times 1 + 3 \times 1$
$= 20$

$I(0,1) = 11 - 1 + 5 - 3$
$= 12$

$I(1,0) = 11 + 1 - 5 - 3$
$= 4$

$I(1,1) = 11 - 1 - 5 + 3$
$= 8$.

$I(r,c) = \begin{bmatrix} 20 & 12 \\ 4 & 8 \end{bmatrix}$
↓ processed

↓ different than $\begin{bmatrix} 5 & 3 \\ 1 & 2 \end{bmatrix}$

Solution to →:

(i) $k = 1$; $k' = \frac{1}{4}$
(ii) $k = \frac{1}{4}$; $k' = 1$
(iii) $k = k' = \frac{1}{2}$

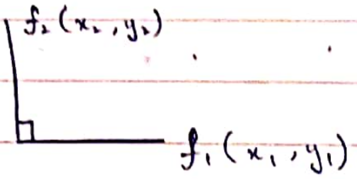
fourier trans. : k
inverse " : k'

Date

freq. coefficient types.

Orthogonal freq.

vectors when at 90° to eachother

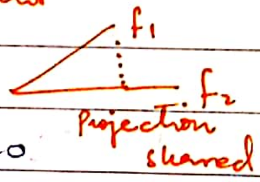


→ That means chosen freq. should be unique + don't share anything hence, giving unique coefficients.
If not unique then you can't get original image back because now there are multiple projections.

$|f_1||f_2| \cos 90 = 0$ → if 0 then orthogonal

$x_1x_2 + y_1y_2 = 0$

means: vectors don't share anything. imp. because now we get unique coefficients, not shared by any other vector



$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \quad 1-1+1-1=0$$

Orthonormal freq.

frequencies that are orthogonal and have a magnitude equal to one.

$f = [4 \ 5]$

$|f| = \sqrt{4^2+5^2} =$

$M_1 = \sqrt{1^2+1^2+1^2+1^2} = \sqrt{4} = 2.$

1st freq. basis

- choosing $k = k' = \frac{1}{2}$, we are normalizing the '2' in orthonormal. to make it 1.

Date

Q write base freq. that are orthogonal + orthonormal ?

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

or

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Apply both on

12	15
14	17

$k = k' = 1$ (because normalization is already done)

$$T(u,v) = k \cdot \sum_{r=0}^{1} \sum_{c=0}^{1} I(r,c) \cdot B(r,c,u,v)$$

$$T(0,0) = 12 \times \frac{1}{2} + 15 \times \frac{1}{2} + 14 \times \frac{1}{2} + 17 \times \frac{1}{2} = 6 + 7.5 + 7 + 8.5 = 29$$

$$T(0,1) = 12 \times \frac{1}{2} - 15 \times \frac{1}{2} + 14 \times \frac{1}{2} - 17 \times \frac{1}{2} =$$

$T(1,0) =$

$T(1,1) =$

Now inverse + check. you should get

12	15
14	17

. ✓

~~F(u,v) = k $\sum_{r=0}^{N-1}\sum_{c=0}^{N-1}$ I(r,c) . B(r,c)~~

Date

pixel/coordinate freq. can be represented as both — F(u,v)

NOTE:

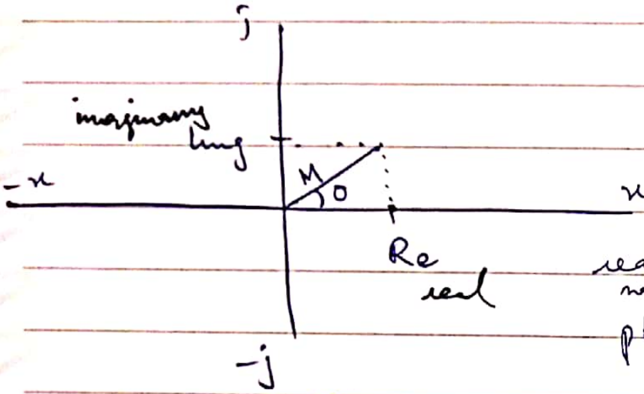
Rectangular + Polar Coordinate:

ways to represent no. : ① Rectangular coordinate

$Re + j\,Img$

② Polar Coord system

$Me^{j\theta}$



real axis: magnitude: $M = \sqrt{Re^2 + Img^2}$

phase : $\theta = \tan^{-1}\left|\frac{Img}{Re}\right|$

Rectangular Representation

$$F(u,v) = \frac{1}{MN}\sum_{r=0}^{M-1}\sum_{c=0}^{N-1} F(c,r)\, e^{-j2\pi[(cu+rv)/MN]}$$

$I_c = [\,3 \quad 2 \quad 2 \quad 5\,]$

$e^0 = 1$

$$F(u) = \frac{1}{N}\sum_{c=0}^{N-1} I(c)\cdot e^{(-2\pi j u c)/N}$$

$F(0) = \frac{1}{4}[3\times 1 + 2\times 1 + 2\times 1 + 5\times 1] = 3 + 0j$

$F(1) = \frac{1}{4}[3(1) + 2(-j) + 2(-1) + 5(+j)]$

$= \frac{1}{4}[1 + 3j] = \frac{1}{4} + \frac{3}{4}j$

$F(2) = \frac{1}{4}[3(1) + 2(-1) + 2(+1) + 5(-1)]$

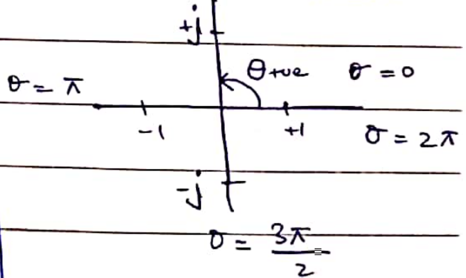
$= -\frac{1}{2} + 0j$

all no. are complex in fourier

$F(3) = \frac{1}{4}[3(1) + 2(+j) + 2(-1) + 5(-j)]$

$= \frac{1}{4} - \frac{3}{4}j$.

$\theta = \frac{\pi}{2}$ $\boxed{e^{j\theta}}$



$e^{j\theta}$ $\theta = \frac{\pi}{2}$

$e^{j\theta} = +j$

$\theta = \frac{2\pi}{3}$ e

$e^{\pi j}$ $e = -1$

$e^{\frac{3\pi}{2}j} = -j$

$3+0j$	$\frac{1}{4}+\frac{3}{4}j$	$-\frac{1}{2}+0j$	$\frac{1}{4}-\frac{3}{4}j$

low pass : ~~sharpening~~ smoothing (remove noise)
high pass : sharpening , edge detection

Date December 2nd, 2021.

Rectangular Coords

$I(c) = [3 \quad 2 \quad 2 \quad 5]$

$F(u) = \begin{bmatrix} 3+0j & \frac{1}{4}+\frac{3}{4}j & -\frac{1}{2}+0j & \frac{1}{4}-\frac{3}{4}j \end{bmatrix}$

$$I(c) = \sum_{u=0}^{N-1} F(u) \cdot e^{\frac{2\pi j u c}{N}}$$

$F(u) = [\, 3 \,,\quad \,,\quad \,,\quad \,]$

$M = \sqrt{Re^2 + Img^2}$
$= \sqrt{3^2 + 0}$
$= 3$

$\theta = \tan^{-1}\left(\frac{0}{3}\right) = 0$

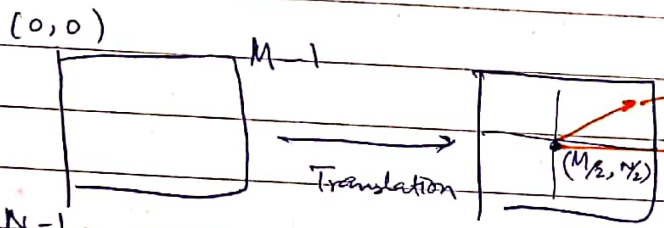
Polar coord $= 3e^{j0} = 3$

$M = \frac{\sqrt{10}}{4}$

$\theta = 0.32175$

$\frac{\sqrt{10}}{4} e^{j0.322}$

"if you don't $\times$ by $\frac{1}{4}$ in forward then you $\times$ while inversing



away from center : ~~high~~ low freq.

Amplitude Profile

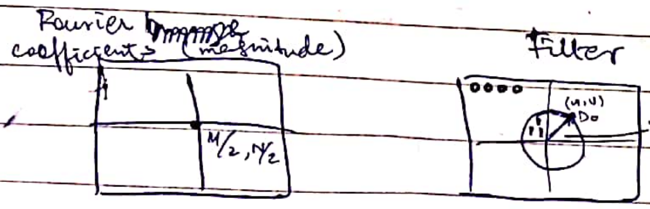
$M = \sqrt{Re^2 + Img^2}$

high contrast img? high freq. sharp img

low freq img? smooth img

(i) Low Pass Filter :-

$H(u,v)$ is a binary with radius Do



allow only Do radius freq.

$$H_{low}(u,v) = \begin{cases} 0 & ; D(u,v) > D_0 \\ 1 & ; D(u,v) \le D_0 \end{cases}$$

distance b/w origin and pixel

pixels' coordinate in Do radius

Date ↑

threshold

$$D(u,v) = \sqrt{\left(u-\frac{M}{2}\right)^2 + \left(v-\frac{N}{2}\right)^2} \qquad 0 \leq D_0 \leq N/2$$

if M×N image: then smaller one

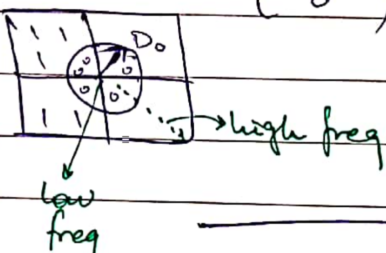
if M small: $\frac{M}{2}$

if N small: $\frac{N}{2}$

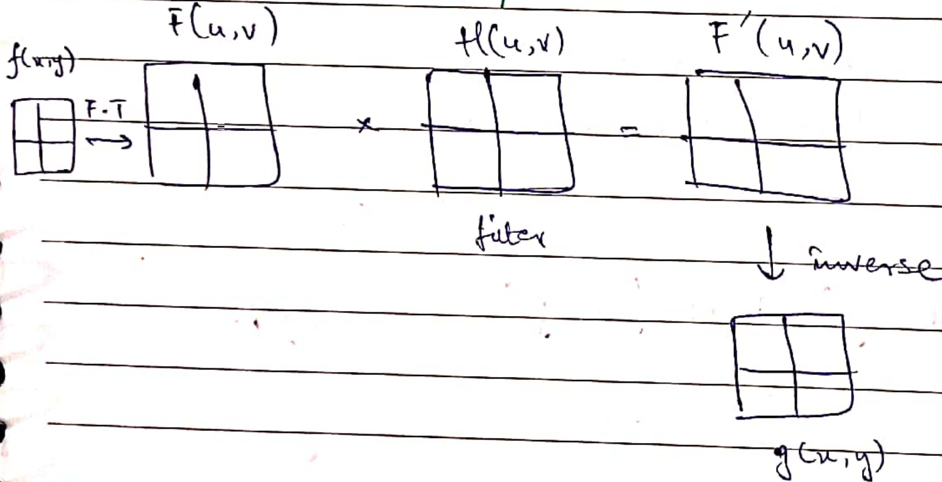
when $H_{high} = 1 - H_{low}(u,v)$

$$H_{high}(u,v) = \begin{cases} 1 & ; \ D(u,v) > D_0 \\ 0 & ; \ D(u,v) \leq D_0 \end{cases}$$

Sharpening, high pass
Smoothing, low pass



* when radius ↑ allowing more high freq. so sharpening



* Image Enhancement
- remove noise
- highlight detail
- highlight a part of img

Noise Removal

6th December, 2021

Image Restoration:

→ Image restoration is the process of removing noise (Additive/Multiplicative) from the images

→ Image restoration is different from image enhancement in the sense that it is more objective process, while enhancement is subjective

→ noise model creation → noise removal

↓ MRI image example

judges perceive it differently

~~Sir sees uniform so~~ Sir applies mean F. I apply median F

judges see that gaussian noise

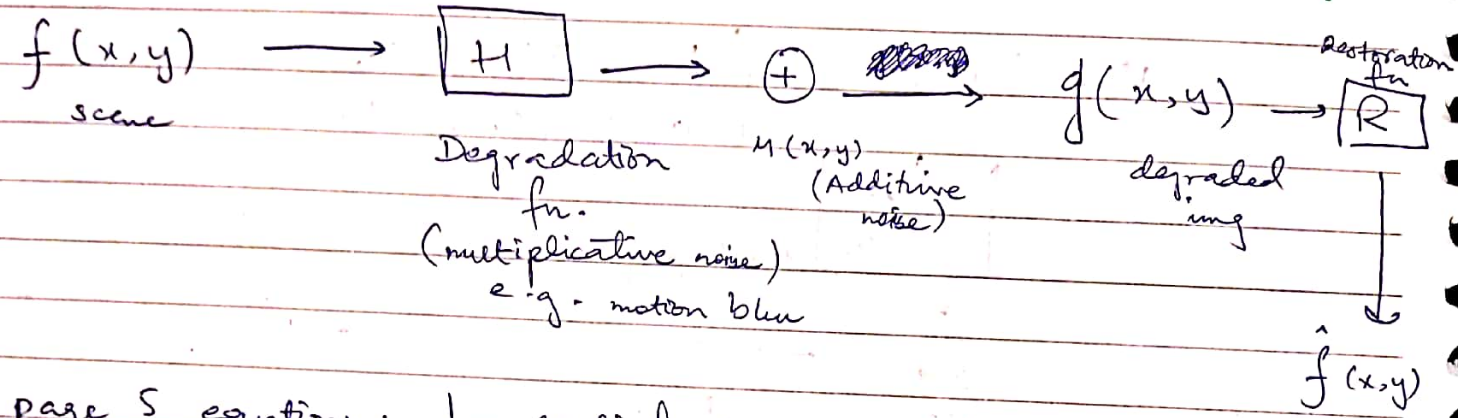
model noise → check type of noise → remove noise using filter

Date

* objective : Prior knowledge
  subjective : heuristic process

we choose R such that $\hat{f}(x,y)$ is very close to $f(x,y)$

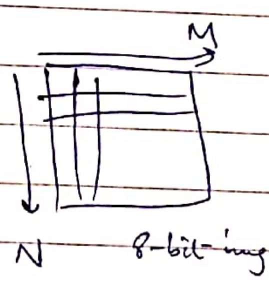
Image Restoration Model:



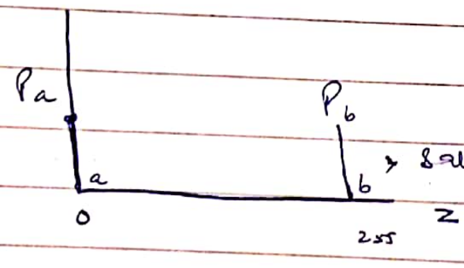
page 5 equations : 1 – spatial domain eq.
2 – freq. domain eq.

Noise Model:

$$g(x,y) = f(x,y) + \eta(x,y)$$



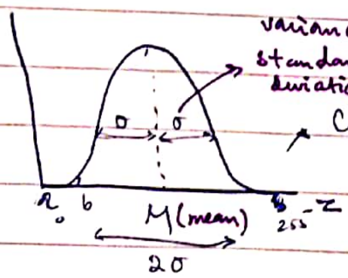
additive noise



→ salt & pepper noise

$$P(z) = \begin{cases} P_a & ; z = a \\ P_b & ; z = b \\ 0 & ; \text{elsewhere} \end{cases}$$

P(z) = prob. of z



→ Gaussian noise model :

$$P(z) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} e^{-(z-\mu)^2/2\sigma^2} & ; z \geq 0 \\ 0 & ; z < 0 \end{cases}$$

$$\mu = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i,j)$$

Scanned with CamScanner

Date

Rayleigh Noise: for ultrasound, MRI, seabed images.

Uniform N:

Impulse N. (salt & pepper)

* How to model noise? → camera, ~~mobile camera, MRI machine~~, XRay mach

(1) you have the detector.

or (2) if detector's technical specifications are available.

or (3) if only images are available

(i) look for a smooth area and extract a patch

(ii) draw histogram of image

(iii) calculate ~~mean~~ μ and σ.

Dec 9th, 2021.

Static Filters: • fixed size e.g. 3x3 mean filter

• have same behavior irrespective of image characteristics

$$\begin{bmatrix} 3 & 3 & 3 \\ 3 & 255 & 3 \\ 3 & 3 & 3 \end{bmatrix}$$

• well suited when prob. of noise < 0.2. $P_{noise} < 0.2$

Dynamic Filters: • img changes according to img characterist

• show different behavior

• $P_{noise} > 0.2$, well suited

Scanned with CamScanner

Date ~~December 2022, 2022.~~ ~~Monday~~

Static Filters :

(i) Mean :

$$g(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} f(s,t)$$

(ii) Geometric Mean: → Loses less img detail as compared to mean filter.

$$g(x,y) = \left[\prod_{(s,t)\in S_{xy}} \left(f(s,t) \right) \right]^{\frac{1}{mn}}$$

Difference in both:

$$\text{mean} = \frac{3\times 8 + 255}{9} = 31$$

$$G.M. = (3^{8}\times 255)^{\frac{1}{9}} = 4.9 \approx 5 \text{ (better)}$$

- G.M. don't work if img has even one 0 pixel value

$$\begin{bmatrix} 3 & 3 & 3 \\ 3 & 3 & 255 \\ 3 & 0 & 3 \end{bmatrix}$$ × not applicable cuz anything multiplied by 0 will be 0.

(iii) Harmonic Mean Filter:

$$g(x,y) = \frac{mn}{\sum_{(s,t)\in S_{xy}} \frac{1}{f(s,t)}}$$

good for removing salt + pepper noise

$$= \frac{9}{\frac{1}{3} + \frac{1}{3} \ldots}$$

Date

(iv) Contra Harmonic Mean:

$$g(x,y) = \sum_{(s,t)\in S_{xy}} \left[\frac{f(s,t)^{Q+1}}{f(s,t)^{Q}} \right]$$

3	3	3
3	255	3
3	3	3

↑ Salt noise

for salt noise (255) $Q = -ve$ (any -ve value)

for pepper noise (0) $Q = +ve$

$Q = -2$

$$f(1,1) = \frac{3^{-2+1} + 3^{-2+1} + \dots + 255^{-2+1}}{3^{-2} + 3^{-2} + \dots + 255^{-2}} \approx 3.$$

(v) Alpha trimmed Mean: for multiple types of noises.

(linear / non L).

sorted img: 3 3 3 3 3 3 3 3, 255 (med: the middle 3) $\quad 0 \le \alpha \le \frac{d}{2}$

In 3x3, $mn = 9$ $\quad d = 8$. $\quad d = mn - 1$

$$0 \le \alpha \le \frac{8}{2} = 4$$

- if we choose $\alpha = 4$, then it trims first + last 4 pixels and acts as median filter (non linear). $= 3$
- if $\alpha = 1, 2$ or 3, then it trims and acts as mean (linear).

$\alpha = 2$ $\quad$ mean $= \frac{3+3+3+3+3}{5} = \frac{15}{5} = 3.$

Dec 13^{th}, 2021

Adaptive Filters Mean

(i) local noise reduction:

$$\begin{matrix} 3 & 3 & 3 \\ 3 & 255 & 3 \\ 3 & 3 & 3 \end{matrix}$$

$$\underset{\text{variance}}{\delta^2} = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} (f(s,t) - \mu)^2$$

$$\mu = 31$$

$$\delta^2 = \frac{1}{9} \left[(3-31)^2 + (3-31)^2 \ldots (255-31)^2 \ldots \right]$$

$\sqrt{\delta^2}$ ← standard deviation.

$$g(x,y) = f(x,y) - \frac{\delta_n^2}{\delta_L^2} [f(x,y) - m_L]$$

$f(x,y)$: whole img

δ_n^2: whole img noise, global

δ_L^2: local noise patch

m_L: local patch mean

Case 0: $g(x,y) = 255 - 1.2 [255 - 31] = -14$

Case 1: $\delta_n^2 = 0$; same img as ~~~~ same output

$\delta_L^2 = 0$; expand filter size

Case 2: $\delta_n^2 > \delta_L^2$; performs small amount of smoothing

Case 3: $\delta_L^2 > \delta_n^2$; performs smoothing

Case 4: $\delta_n^2 = \delta_L^2$; performs normal smoothing

$$\begin{matrix} 3 & 3 & 3 \\ 3 & 255 & 3 \\ 3 & 3 & 3 \end{matrix}$$

Date **Adaptive Median Filter :**

$Z_{min} = 3$. $Z_{max} = 255$ $Z_{med} = 3$

$S_{xy} = 7 \times 7$ Z_{xy} = value at (x,y) location

Conditions (1) Z_{med} is a noise or not

$Z_{min} < Z_{med} < Z_{max}$

$3 < 3 < 255$ not satisfied

Since, This condition is false we will increase the size of filter.

By increasing size let's say we get

$2 < 3 < 255$

Now this condition is true, Z_{med} is a noise

(2) Z_{xy} is a noise or not

$Z_{min} < Z_{xy} < Z_{max}$

If This condition is False Then replace it with Z_{med}

otherwise go with Z_{xy}. and Z_{xy} is noise.

Date December 16th, 2021 Chapter 5 (book)

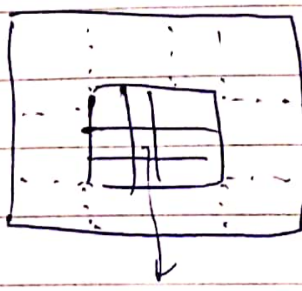
Past lecture

Correction : Local Noise Reduction Filter

$$g(x,y) = f(x,y) - \frac{\sigma_f^2}{\sigma_L^2}\left(f(x,y) - m_L\right)$$

σ_f^2 ← noise

m_L ← local mean



x, y

Case 1: $\sigma_f^2 = 0 \longrightarrow$ output $= f(x,y)$

$\sigma_L^2 = 0 \longrightarrow$ increase filter size;
at max size $o/p = f(x,y)$

variance ↑
noise amount ↑

Case 2: $\sigma_L^2 > \sigma_f^2$; then output a value close to $f(x,y)$
(edge) Hence, preserve the edges

high σ_L^2 is for edges

Case 3: $\sigma_f^2 > \sigma_L^2$; o/p a value that does a little smoothing

Case 4: $\sigma_f^2 = \sigma_L^2$; $o/p = m_L$

normal smoothing

smooth $\sigma_L^2 = 0$

3	3	3
3	3	3
3	3	3

$\sigma_L^2 = 6, \ldots$

3	3	3
3	255	3
3	3	3

$\sigma_L^2 = \ldots$

24	40	200
20	60	240
3	255	180

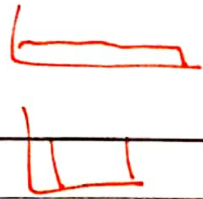
σ_f^2

e.g. σ_L^2 12000 $\sigma_f^2 = 6000$

$$\frac{6000}{12000} = 0.5$$

Scanned with CamScanner

Date New topic Dec. 16. 2021



Morphological Image Processing :

→ we will process shapes to remove unwanted artifacts (e.g. intrusions, extrusions etc).

→ Images are usually the output of a segmentation process, and are in binary format

→ Topics :

(i) Basic operations : (a) Erosion (b) Dilation

(ii) Compound operations : (a) Opening (b) closing

(iii) Morpho logical algorithms : (a) Boundary extraction (b) Region Filling

$f(x,y) =$

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

Binary image

$S =$

0	1	0
1	1	1
0	1	0

all can be 1

Binary Structuring element

(i) (a) Erosion :

$$\hat{f}(x,y) = \begin{cases} 1 ; & \text{if } s \text{ fits } f \\ 0 ; & \text{otherwise} \end{cases}$$

structuring elements fits image patch.

0	0	0
1	1	1
0	1	0

img patch (not fitting) → Hitting: if atleast one pixel in S overlap with one pixel in img patch

0	1	0
1	1	1
0	1	0

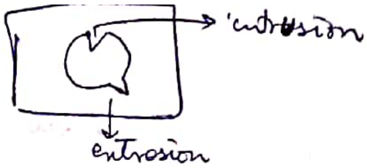
s. element

0	1	0
1	1	1
0	1	0

(fitting) → if all pixels in S overlap with all on pixels in img patch.

1 – obj
0 – bg

$\ominus$ erosion
$\oplus$ dilation



Date

(i) (b) Dilation :

$$\hat{f}(x,y) = \begin{cases} 1 & ; \text{ if } s \text{ hits } f \\ 0 & ; \text{ otherwise} \end{cases}$$

why we do Morphological Image Processing ?

(ii) (a) Opening :

$$\underbrace{f \circ s}_{\text{opening}} = (f \underbrace{\ominus}_{\text{erosion}} s) \overset{\text{dilation}}{\oplus} s$$

(ii) (b) Closing

$$\underbrace{f \bullet s}_{\text{closing}} = (f \underbrace{\oplus}_{\text{dilation}} s) \underbrace{\ominus}_{\text{erosion}} s$$

December 20^{th}, 2021. Monday.

(iii) (a) Boundary Extraction

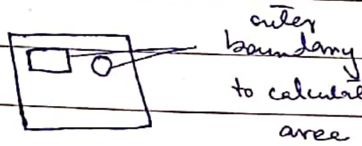
$$\underbrace{B(A)}_{\text{image}} = A - (A \underbrace{\ominus}_{\text{erosion}} \underbrace{S}_{\text{3x3 structuring element}})$$

0	1	0
1	1	1
0	1	0

- gives exact boundary of object.

$$B_{outer}(A) = (A \underbrace{\oplus}_{\text{dilation}} S) - A$$

- gives outer boundary of the object



to calculate length or area of shapes

(iii) (b) Region Filling :

– semi automatic

$A =$

1	1	1	0	0
1	0	0	1	0
1	0	0	1	0
0	1	0	1	0
0	1	1	0	0

$X_0 =$
c direct

0	0	0	0	0
0	1	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

highlighted region array

in X_0 we tell it what pixel to fill = semi-auto

$$X_k = (X_{k-1} \oplus S) \cap A^c$$

$k = 0, 1, 2, \ldots$

Date

Stop when $X_k = X_{k-1}$ and the image B, with filled ~~...~~ region will be:

$$B = A \cup X_k$$

$$X_1 = (X_0 \oplus S) \cap A^c$$
$$X_2 = (X_1 \oplus S) \cap A^c$$
$$X_3 = (X_2 \oplus S) \cap A^c$$

$S =$

0	1	0
1	1	1
0	1	0

$X_0 \oplus S =$

0	1	0	0	0
1	1	1	0	0
0	1	0	0	0
0	0	0	0	0
0	0	0	0	0

$X_1 =$

	1	1		
	1			

$X_1 \oplus S =$

0	1	1	0	0
1	1	1	1	0
1	1	1	0	0
0	1	0	0	0
0	0	0	0	0

$X_2 =$

	1	1		
	1	1		

$X_2 \oplus S =$

	1	1		
1	1	1	1	
1	1	1	1	
	1	1		

$X_3 =$

	1	1		
	1	1		
		1		

$X_3 = X_2$? No, so don't stop. Compute X_4

$X_3 \oplus S =$

0	1	1	0	0
1	1	1	1	0
1	1	1	1	0
0	1	1	1	0
0	0	1	0	0

$X_4 =$

	1	1		
	1	1		
		1		

$X_4 = X_3$ so stop.

* laplacian is sensitive to ~~[illegible]~~ isolated points

Date

Image Segmentation :

↓

process of identifying boundaries of objects

- 3 types of discontinuities :
  - Points
  - Lines
  - Edges

(i) Points :

Laplacian ⟶ $\nabla^2 f = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$

↓

edge profile (contains both -ve / +ve)

↓

normalize

↓ $f(x,y)$

threshold for point detection

$$f'(x,y) = \begin{cases} 1 ; & f(x,y) \geq T \\ 0 ; & f(x,y) < T \end{cases}$$

- gives the highlighted point in image
- when threshold set high, only points will come, those points will give double response than other points so they get highlighted

0	0	255	0	0	0

$f'(x)$ 0 255 -255 0 0

$f''(x)$ 255 -510 255 0

$\underbrace{-510}_{\text{double response}}$

Thick edges

↑

1st O.D. non zero value along ramps

2nd O.D. 0 along ramps

~~LoG~~ LoG : better cuz first removes noise then applies laplacian

→ Gaussian is smoothing filter, removes noise

Date

(ii) Lines:

operator

$$H_E = \begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} \qquad H_V = \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

horizontal edge — vertical |

$$H_{E(+45°)} = \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \qquad H_{E(-45°)} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

/ \

(iii) Edges: • set of connected pixels that lie on the boundary b/w two regions.

• transition

• 1st O.D. gives thick edges
2nd O.D. sensitive to minute details, isolated points

2 Detectors : (a) Canny (b) Laplacian of Gaussian (LoG)

(a) Canny: 1st O.D. + Gaussian

$$\nabla G(x,y) = G'_x(x,y) + G'_y(x,y)$$

$$= e^{\frac{-(x^2+y^2)}{2\sigma^2}} \cdot \frac{d}{dx}\left(\frac{-x^2+y^2}{2\sigma^2}\right) -$$

$$e^{\frac{-(x^2+y^2)}{2\sigma^2}} \cdot \frac{d}{dy}\left(\frac{-(x^2+y^2)}{2\sigma^2}\right)$$

↓ after adding

$$\nabla^2 h(r) = -\left[\frac{r^2-\sigma^2}{\sigma^4}\right]\exp(-r^2/2\sigma^2)$$

(b) LoG: 2nd O.D. + Gaussian

$$G(x,y) = e^{\frac{-(x^2+y^2)}{2\sigma^2}}$$; where σ is std. $\sigma = 1$

(filter in slide 23)

we don't use laplacian alone. we use it w/ gaussian

- Img Restoration (noise models, modeling, restoration filter)
- 2D Fourier Transformation
- Morphological Img Processing
- Img Segmentation.

Date December 23rd, 2021.

Laplacian of Gaussian

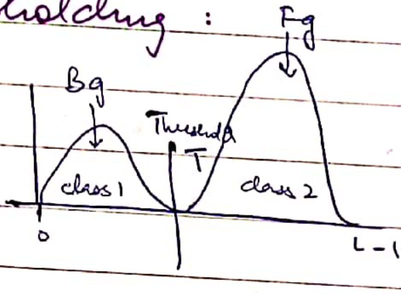
- 2nd O.D. w/ smoothing filter

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

5×5 filter

Image Segmentation using Thresholding:

(i) Thresholding:



$$g(x,y) = \begin{cases} 1 \; ; & f(x,y) \geq T \\ 0 \; ; & f(x,y) < T \end{cases}$$

bi_modal (having 2 classes)

given the histogram, find T

(ii) Global Thresholding algorithm:

$\Delta T_E = 2$

estimated change in threshold

10	10	10
10	24	255
255	255	255

Img

Step ①: Choose a random value of T b/w [0 — L-1] and divide the image into 2 regions G_1 & G_2

SOP

BW — Stuttgart

Date

Step ② : Calculate the avg. intensity of G_1 and G_2.
i.e. m_1 + m_2
m_1 → avg. intensity of G_1; m_2 → avg. int. of G_2

Step ③ : Calculate new Threshold $T_{new} = \frac{m_1 + m_2}{2}$ and calculate $\Delta T = T_{new} - T$

Step ④ : If ΔT is greater than ΔT_E, repeat step ② - ④

Example:

$T = 8$ $m_1 = 0$ $m_2 = 120$

$T_{new} = \frac{m_1 + m_2}{2} = \frac{120}{2} = 60.$

new T

Now $T_{new} = 60$ $\Delta T = 60 - 8 = 52$

$G_1 = \{10, 10, 10, 10, 24\}$ $G_2 = \{255, 255, 255, 255\}$

$m_1 = 12.8 \approx 13$ $m_2 = 255$

$T_{new} = \frac{13 + 255}{2} = 134$

$\Delta T = 134 - 52 = 82$

$G_1 = \{10, 10, 10, 10, 24\}$ $G_2 = \{255, 255, 255, 255\}$

$m_1 = 13$ $m_2 = 255$

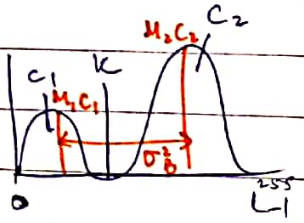
T value 134. $T_{new} = 134$, $\Delta T = 0$, $0 < 2$ so $T = 134$ (DIY)

Scanned with CamScanner

Date

Otsu's thresholding Algorithm:

- operates not on images, but histogram of images
  ↓
  bi-modal
  2 classes



- pixels
- bits
- 1D array

9	9	9
9	9	9
9	9	9

$$m = \sum_{i=0}^{L-1} i \cdot P(i) \; ; \rightarrow \text{on image}$$

avg. intensity

$$m = 9$$

$$T(k) = k$$

$$M(C_1) = \sum_{i=0}^{k} \frac{i \cdot P(i)}{P_{C_1}} \rightarrow \text{on portion of histogram}$$

$$M(C_2) = \sum_{i=k+1}^{L-1} \frac{i \cdot P(i)}{P_{C_2}}$$

prob of a shade to lie in region C_1 ←

$$P(C_1) = \sum_{i=0}^{k} P(i)$$

$$P(C_2) = \sum_{i=k+1}^{L-1} P(i)$$

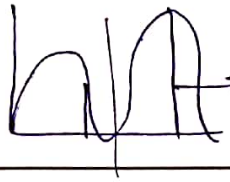
$$P_{Total} = P_{C_1} + P_{C_2} = 1$$

Total variance of Image:

$$\sigma_T^2 = \frac{\sum_x \sum_y \left[f(x,y) - M_{f(x,y)} \right]^2}{M \times N}$$

→ mean of whole image

Date



if we put this bar in C_1 and calculate variance ~~the variance will be~~ higher.

Otsu ↓

$$\boxed{\sigma_T^2} = \underbrace{\sigma_W^2}_{\substack{\text{within}\\\text{Class variance}\\\text{(minimize)}}} + \overbrace{\sigma_B^2}^{\substack{\text{between class}\\\text{variance}\\\text{(maximize)}}}$$

* choose T that will minimize + maximize

class 1 pixels corelated

" 2 " "

- $\sigma_W^2 = P_{C_1} \cdot \sigma^2_{C_1} + P_{C_2} \cdot \sigma^2_{C_2}$

$$\sigma_{C_i}^2 = \frac{\sum_{i=1}^{m} (P(i) - M(C_i))^2 \cdot i}{P_{C_i}}$$

$T = 0, 1 \ldots\ldots L-1$

$T(k) = k$

$\sigma_W^2(k) = ?$ pick a T which is minimum.

calculate for 255 values if 8-bits
(k)

- $\sigma_B^2(k) = P_{C_1} \cdot P_{C_2} \, (M(C_1) - M(C_2))^2$

End