7/9/18
Frequency
Domain | Space

→ Frequency means describing the character

→ A Signal (wave in time or image in Spatial domain) can be decomposed or seperated into a sum of Sinusoids of different frequencies, amplitude and phase.

$$\underset{\text{amplitude}}{A} \sin(\omega t + \underset{\text{phase}}{\phi})$$

→ 1D Domain Example:

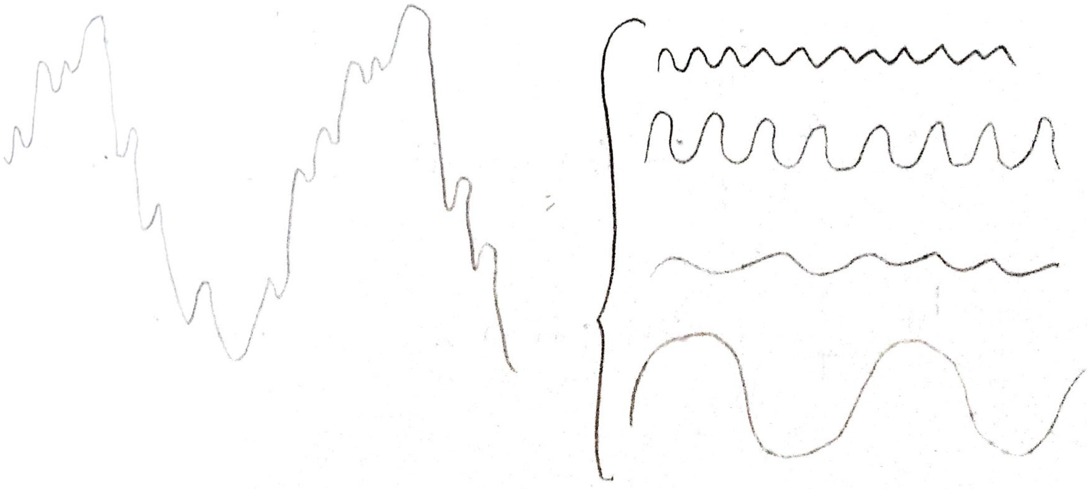
Consider a Complicated Sound played in a piano (or) guitar.

Frequency domain:

as a set of spatial frequency values / component (represented as 2D grid of spatial)

→ If an image represented in frequency Space has high frequencies then it means the image has sharper edges.

Any function that periodically repeats itself can be expressed as a sum of Sines and cosines of different frequencies each multiplied by a different coefficient. a fourier serie.



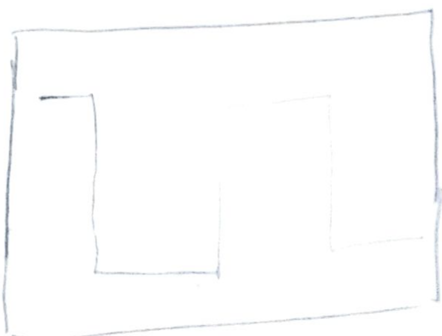
Our building block.

$$A \sin(\omega t + \phi)$$

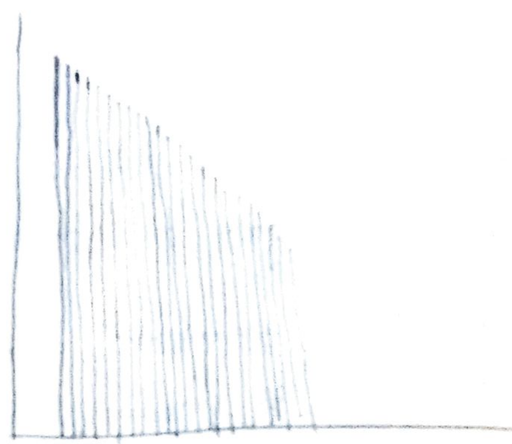
→ Add enough of them to get any signal $f(x)$ you want.

→ Coarse structure — low frequency

→ fine structure — high frequency



$$= A \sum_{k=1}^{\infty} \frac{1}{k} \sin(2\pi k t)$$



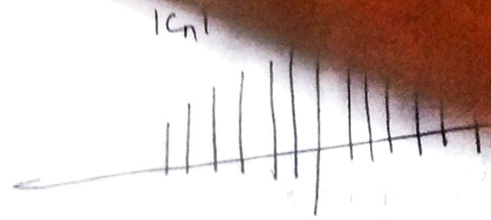
$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + \sum_{n=1}^{\infty} b_n \sin(n\omega_0 t) \qquad \omega_0 = \frac{2\pi}{T}$$

$$a_0 = \frac{2}{T} \int_{t_0}^{t_0+T} f(t)\, dt \qquad f(t) = C_0 + \sum_{n=1}^{\infty} C_n \cos(n\omega_0 t \ldots)$$

$$a_n = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \cos(n\omega_0 t)\, dt \qquad n = 1 \ldots \qquad C_n = \sqrt{a_n^2 + b_n^2}$$

$$\theta_n = \tan^{-1}\left(\frac{b_n}{a_n}\right)$$

$$b_n = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \sin(n\omega_0 t)\, dt \qquad n = 1, 2, 3 \ldots$$




What if the Signal is A-periodic?

→ Yes, and the fouriers transform provides the tool for this analysis

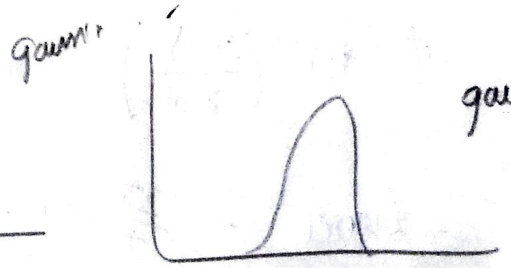
→ A-periodic signal can be treated as periodic with time period tending to infinity in the limit.

$$F(u) = \int_{-\infty}^{\infty} f(x)\, e^{-j 2\pi u x}\, dx$$

$$e^{j\theta} = \cos\theta + j\sin\theta \qquad j = \sqrt{-1}$$

Inverse Fourier Transformation (IFT)

$$f(x) = \int_{-\infty}^{\infty} F(u)\, e^{j 2\pi u x}\, du$$

The Discrete fourier Transforms

Frequency mean intensity variation of image in spac

helps Us to decompose the Signal

faster in frequency domain

Fourier Series & Transformation:

10/9/18

$$F(n) = C_0 + \sum C_n \sin(n\omega_0 x + \phi)$$

$\underbrace{C_n}_{\text{Amplitude}}$ $\underbrace{n\omega_0 x}_{\text{frequency}}$ $\underbrace{\phi}_{\text{phase}}$

$$\underset{\text{Space}}{F(x)} \xrightarrow{FU} \underset{\text{frequeny}}{F(U)}$$

$$F(U) = \int_{-\infty}^{\infty} f(x)\, e^{-j2\pi Ux}\, dx$$ → Eulers formula

Inverse fourier Transfor

IFT $$f(x) = \int_{-\infty}^{\infty} f(U)\, e^{j2\pi Ux}\, dU$$

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

$$F(x) = R(\omega) + jI(\omega)$$

$$|F(x)| = \sqrt{R^2 + I^2}$$

$$\phi = \tan^{-1}\left(\frac{I(\omega)}{R(\omega)}\right)$$

fourier Series - Discrete - $\frac{2\pi}{T}$ → finite

fourier Transform - Continuous - $\frac{2\pi}{T}$ T → infinity

Discrete fourier Transform

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

for

$u = 0, 1, 2, \ldots, M-1$ $\quad O(N^4) \quad MN \times MN$

$v = 0, 1, \ldots, N-1$ $\quad (MN)^2$

Inverse Discrete Fourier transform is given by equation

$$f(x,y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(u,v)\, e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

$D(u,v) = c \log(1 + |F(u,v)|)$ → Normalized

→ Phase carries more info of image than magnitude
↓
carries relative strength of each Sine Component.

Separability

The 2D DFT can be computed using 1D transform Only

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

Kernel is seperable:

$$e^{-j2\pi\left(\frac{ux+vy}{N}\right)} = e^{-j2\pi\left(\frac{ux}{N}\right)} \times e^{-j2\pi\left(\frac{vy}{N}\right)}$$

→ Fast fourier Transform based ~~Based~~ on Divide & Conquer
↓
reduces no. of Computation

MN log MN time

Properties of DFT

## Periodicity - DFT

DFT & its inverse are periodic with period N

$$F(u,v) = F(u+N, v) = F(u, v+N) = F(u+N, v+N)$$

## Translation

Translation are also possible

$$f(x,y) \leftrightarrow F(u,v)$$

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

to show full period, we need to translate the origin of transform at $u = N/2$ (or at $(\frac{N}{2}, \frac{N}{2})$ in 2D)

## Rotation

Rotating $f(x,y)$ by $\theta$ rotates $F(u,v)$ by $\theta$

## Multiplication & Convolution

| Spatial Domain | | Frequency Domain |
|---|---|---|
| Convolution ← $g = f * h$ | $\leftrightarrow$ | $G = FH$ |
| Multiplication ← $g = fh$ | $\leftrightarrow$ | $G = F * H$ |

So we can find $g(x)$ by fourier transform

$$g = f * h$$

$$\downarrow \quad \downarrow \quad \downarrow$$

$$\boxed{IFT} \quad \boxed{FT} \quad \boxed{FT}$$

$$G = F \times H$$

Filtering Image in Frequency Domain

1. Compute $F(u,v)$ the DFT of Image

2. Multiply $F(u,v)$ by a filter function $H(u,v)$

3. Compute the inverse DFT of the result

Pre Processing → Fourier Transfer $O(N^2 \log N)$ → Filter Function $H(u,v)$ $O(N^2)$ → Inverse fourier transfo $O(N^2 \log N)$ → Enhanced image

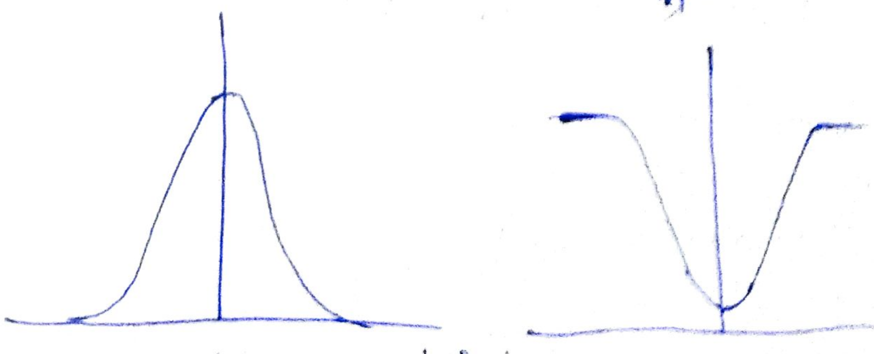
$f(x,y)$

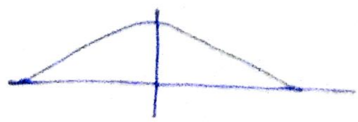
Pre Processing → Spatial filtering $O(N^4)$ → Enhanced image ↓ $g(x,y)$

Smoothing / Blurring results from:

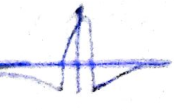
- Pixel averaging in the spatial domain
  - Each pixel is the output is a weighted average of it neighbours
  - Is a Convolution whose weight matrix sums to 1
- Low pass filtering in the frequency domain
  - High frequencies are diminished or eliminated

$H(u)$ $\quad H_{hp}(u,v) = 1 - H_{lp}(u,v)$





$$\begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix}$$



$$H(u,v) \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

Symmetric

$F(u,v) \cdot [F(u, v)]$

$$D(u,v) \left[(u - N/2)^2 + (v - N/2)^2\right]^{1/2}$$

The ideal Band pass filter given as

$$H(u,v) \begin{cases} 0 & \text{if } D(u,v) \le D_0 - W/2 \quad W - \text{bandwidth} \\ 1 & \cancel{D(u,v) \le D_0 \le D(u,} \; D_0 - W/2 \le D(u,v) \le D_0 + W/2 \\ 0 & D(u,v) \ge D_0 + W/2 \end{cases}$$

Gaussian low pass filter

$$H(u,v) : e^{-D^2(u,v)/2D_0^2}$$

$D_0$ : (Deviation of gaussian [illegible]

Noise - High frequency enhanced

19/9/18

<u>Image Restoration</u>:

$$H[a F_1(x,y) + b F_2(x,y)] = a H[F_1(x,y)] + b H[F_2(x,y)]$$

$$g(x,y) = h(x,y) * F(x,y)) + \eta(x,y)$$

if position-invariant

$$H[F(x,y)] : h(x,y) * F(x,y) \rightarrow \text{Convolution}$$

$$G(u,v) : H(u,v) \times F(u,v) + N(u,v)$$

If only noise (

* additive noise

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

Multiplicative nois

$$g(x,y : f(x,y \times \eta(x,y)$$

$\ln g(x,y) = \ln F(x,y) + \ln n(x,y)$

Additive noise:

Arithmetic mean: $\hat{f}(x,y) = \dfrac{\sum_{(s,t)\in S_{xy}} g(s,t)}{mn}$

Geometric mean:

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

Harmonic mean:

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

works for Salt noise, but fails for pepper noise

Also does well with Gaussian noise

Contraharmonic mean:

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

+ve Q values - pepper noise

-ve Q value - Salt noise

20/9/18

Max filter good for ~~Salt~~ pepper noise & min filter good for Salt noise

<u>Max filter</u> $\hat{f}(x,y) = \max_{(s,t)\in S_{xy}} g(s,t)$

<u>Min filter</u> $\hat{f}(x,y) = \min_{(s,t)\in S_{xy}} g(s,t)$

<u>Median filter</u>

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{\text{median}} \{g(s,t)\}$$

Good for Salt & pepper noise if both are present

<u>Alpha-trimmed mean filter:</u>

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

We can delete $d/2$ lowest & $d/2$ highest grey levels

So $g_r(s,t)$ represents the remaining pixels

Used for removing both Gaussian & Salt n pepper the same time

<u>Adaptive Filter:</u>

The behaviour of adaptive filters changes depending on the characteristics of the image inside the filter region

Adaptive local noise reduction filter.

- Response based on 4 quantities
  - Local variance, Variance of noise, $g(x,y)$ & local mean

Behaviour of the filter

- If variance of noise of zero, returns $g(x,y)$
- If local variance is high compared to variance of noise, return a value to close to $g(x,y)$
- If the two variances are equal, return the mean value

$$\hat{f}(x,y) = g(x,y) - \frac{\sigma_n^2}{\sigma_L^2}\left[g(x,y) - m_L\right]$$ local mean

$S_{xy} \to m_L$

$S_{xy} \to \sigma_L^2 >> \sigma_n^2$

$\sigma_L^2 \cong \sigma_n^2$

Different operations for different neighbours

## Adaptive Median Filtering:

- Remove spatially dense impulse noise
- Reduce distortion

Level A: $A_1 : Z_{med} - Z_{min}$

$A_2 : Z_{med} - Z_{max}$

If $A_1 > 0$ and $A_2 < 0$ Go to level B

Else increase the window size

If window size $\leq S_{max}$ repeat level A

Else Output $Z_{med}$

Level B: $B_1 : Z_{xy} - Z_{min}$

$B_2 : Z_{xy} - Z_{max}$

If $B_1 > 0$ and $B_2 < 0$, output $Z_{xy}$

Else Output $Z_{med}$

Periodic Noise Removal:

→ uses band Reject filter in the frequency domain

Notch Filter:

22/9/18

1) Linear, Position-invariant degradation model

$$g(x,y) : f(x,y) * h(x,y) + \eta(x,y)$$

$$G(x,y) : F(x,y) H(x,y) + N(x,y)$$

First estimate the degradation function
then find the inverse of it
Estimating the degradation function:
→ Find the strong signal content (negligible noise) in the image

$\therefore \eta(x,y) : 0$

→ Esitmate the original image in the window

$$H_s(u,v) : \frac{G_s(u,v)}{\hat{F}_s(u,v)}$$

$G_s(u,v)$ → known
$\hat{F}_s(u,v)$ → estimate
↓
Can be estimated by using the characteristic of the image

if $G_s$ : blurred $F_s$ : deblurred.

By experimentation : If the image acquisition system is available, then obtain the impulse response

- $G(u,v) : H(u,v) F(u,v) + N(u,v)$

$H(u,v) = G(u,v)/A$

Fouriers transform of strong impulse is constant value.

By modeling

ex: Atmospheric turbulence

$$H(u,v) = e^{-k(u^2+v^2)^{5/6}}$$

Derive a mathematical model: ex. Motion blur

$$g(x,y) = \int_0^T f\left((x - x_0(t), y - y_0(t)\right) dt$$

Planar motion

fourier Transform

$$G(u,v) = F(u,v) \int_0^T e^{-j2\pi[u x_0(t) + v y_0(t)]} dt$$

$$H(u,v) = \int_0^T e^{-j2\pi[u x_0(t) + v y_0(t)]} dt$$

degradation

5) Inverse filtering

Compute the estimate by simply dividing the transform of degraded image by the degradation function

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

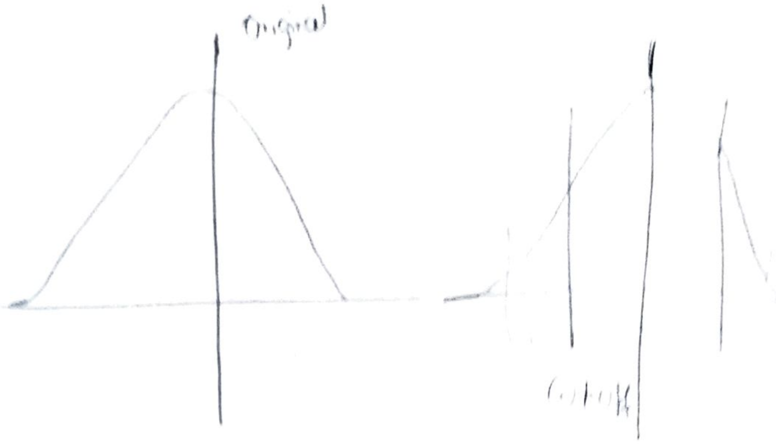
unknown noise

Estimate of original image

Problem (0 (or) small values)

Suffers from noise amplification

(So first reduce the noise by any of the method Additive, Mean (or) Median) limit the Inverse filtering ~~bay~~ vs by using the cut off




Wiener Filter:

→ Incorporate both the degradation & Statistical characteristics of noise

→ Objective Function: find an estimate of $f$ such that the mean Square error between them is minimised

$$e^2 = E\{(f - \hat{f})^2\}$$

$$\hat{F}(u,v) = \frac{H^*(u,v)\, S_f(u,v)}{S_f(u,v)\,|H(u,v)|^2 + S_n(u,v)}\, G(u,v)$$

$S_f$:

$S_n(u,v) = |N(u,v)|^2$ → Power Spectrum of Noise

$$\hat{F}(u,v) = \frac{1}{H(u,v)} \frac{H(u,v)^2}{|H(u,v)|^2 + \underbrace{S_n(u,v)}_{\text{Constant}} / \underbrace{S_f(u,v)}_{\text{Unknown}}}\, G(u,v)$$

$$\hat{F}(u,v) = \frac{1}{H(u,v)} \frac{H(u,v)^2}{H(u,v)^2 + K}\, G(u,v)$$

· Signal to Noise ratio

$$SNR: \frac{\sum^{m}\sum^{n}|F(x,y)|^2}{\sum^{m}\sum^{n}|N(x,y)|^2}$$ should be less

· Mean Square error

$$MSE: \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}|\underbrace{\hat{F}(i,j)}_{\text{reconstructed}} - \underbrace{I(i,j)}_{\text{Original image}}|^2$$ should be more

· Root Mean Square Error

$$RMSE: \sqrt{MSE}$$ should be less

· Peak Signal to Noise Ratio

$$PSNR: 20\log_{10}\left|\frac{255}{RMSE}\right|$$ should be more

24/9/18

Morphological Image Processing

→ Morphological image processing describes range of techniques for extracting image components that are useful in the representation & description of region.

Language of morphological processing is set theory.

$S \in Z^2$ (2 dimentional)

$(x, y) \rightarrow$ 2 dimentional tuple
Spatial location of the pixel

We also use $Z^3$

$(x, y, I) \rightarrow$ Intensity of Image at that pixel.

· Reflection and Translation

$$\hat{B} = \{ w \mid w \in -b, \text{ for } b \in B \}$$

$$(A)_z = \{ c \mid c \in a+z, \text{ for } a \in A \}$$

Reflectio

$(x,y) \rightarrow (-x,-y)$ depending on the origin

$\wedge \rightarrow$ Reflection

$(A)_z$ denotes translation

↓

shifted the object by z

Structuring Elements:

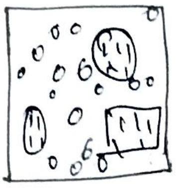
→ structuring elements are ~~set~~ small sets / sub-images used to probe an image under study.

→ For each SE define its origin

→ Shape & size must be adapted to geometric properties for the objects

- For simplicity we will use rectangular structuring element with their origin at the middle pixel.

I



→ But sometime it helps to take non symmetric structuring elements.

Fit :

All on pixels in the structuring element cover on pixels in the image

Hit :

Any on pixel in the structuring element cover on pixels in the image.

Fundamental Operations:

→ Very much like Spatial filtering

Erosion operation:

Erosion of image A by Structuring Element B represented as sets in $Z^2$, is defined as

$$A \ominus B = \{ z \mid (B)_z \subseteq A \}$$

$$g(x,y) = \begin{cases} 1 & \text{if } B \text{ fits } A \\ 0 & \text{otherwise} \end{cases}$$

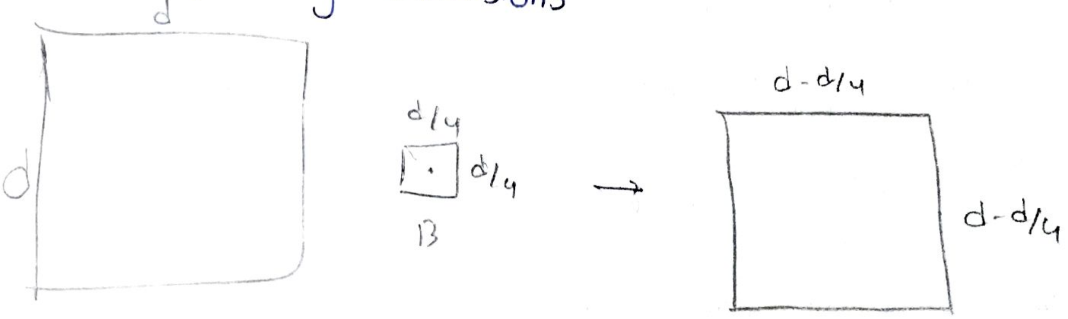
(remove irrelevant properties)

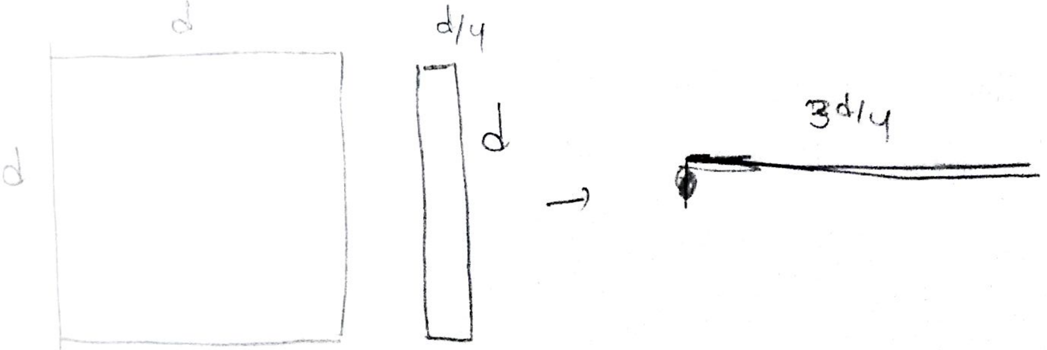
~~What is~~

What is erosion for?

→ Erosion can split apart joined objects

→ Erosion can strip away extrusions



27/9/18

2) Dilation Operation:

Dilation of image A by structuring elemet B is given by $A \oplus B = \{ z \mid (\hat{B})_z \cap A \neq \phi \}$ → reflection of B

The set of all displacements z such that $\hat{B}$ and A overlap by at least one element.

$$g(x,y) = \begin{cases} 1 & \text{if } B \text{ hits } A \\ 0 & \text{otherwise} \end{cases}$$

$\boxed{\text{Why do we do reflection?}}$

If the revese operation of erosion (Converse of each other with respect to set complime.)

What is a dilation for?

→ Dilation can repair breaks

Compound Operations:

Opening:

The opening of image f by structuring element s. denoted by $f \circ s$ is simply an erosion followed by dilation

$$f \circ s = (f \ominus s) \oplus s$$

→ The image is not fully restored.

Closing:

The closing of of image f by structuring element s. denoted by $f \bullet s$ is simply a dilation followed by an erosion

$$f \bullet s = (f \oplus s) \ominus s$$

Morphological Algorithms

→ Boundary extraction
→ Region filling
→ Convex Hull

Boundary Extraction:

The boundary can be given simply as

$$\beta(A) = A - (A \ominus B)$$

Region filling:

Given a pixel inside a boundary, region filling attempts to fill that with object fills (1s).

$$X_k = (X_{k-1} \oplus B) \cap A^c \quad [\text{Until} \quad X_k = X_{k-1}]$$

repeat the operation.

29/9/18

Extraction of Connected Components

$$X_K = (X_{K-1} \oplus B) \cap B$$

1) Remove the noise (opening + closing)

2) Find the boundaries of all the objects Connected Components algorithm.

3) Run the Connected Component algorithm again & find the Object which has more than 2 Connected Components

Hit (or) miss transformation:

$$A \circledast B = (A \ominus X) \cap [A^c \ominus (W-X)]$$

$$B = (X, W-X)$$

27

Corner Detection

erosion

| | | |
|---|---|---|
| x | 0 | 0 |
| 1 | 1 | 0 |
| 1 | 1 | x |

(3)

| | | |
|---|---|---|
| x | 1 | x |
| 1 | 1 | 1 |
| 0 | 0 | x |

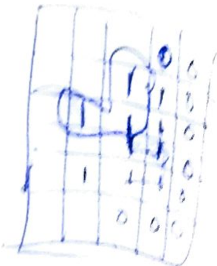
(1)

(2)

| | | |
|---|---|---|
| 0 | 0 | x |
| 0 | 1 | 1 |
| 1 | 1 | x |

(4)

| | | |
|---|---|---|
| x | 1 | x |
| 1 | 1 | 0 |
| x | 0 | 0 |



Convex Hull:

→ A set A is said to be a convex ~~if~~ if the straight line segment joining any two points in A lies entirely in A

Thinning ops

A (1 ⊖ 3)

8/10/18

Cones responsible for Color vision

Color Image Processing

Tristimulus the amount of RGB need to

for any Color (X, Y, Z)

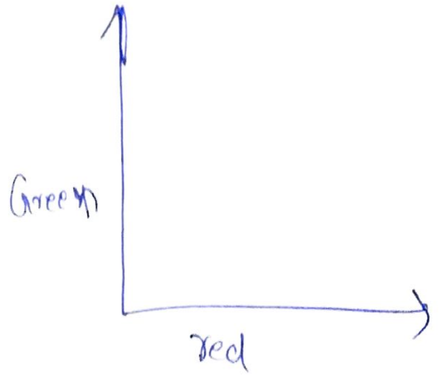
Trichromatic Color x, y, z

$$x = \frac{X}{X+Y+Z}$$

$$y = \frac{Y}{X+Y+Z}$$

$$z = \frac{Z}{X+Y+Z}$$

Chromaticity diagram.




$z = 1-(x+y)$

Point of equal energy

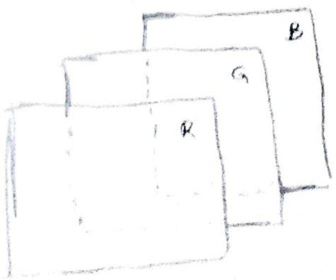
Popular models used in Color image processing

RGB (Red Green Blue)

HSI (Hue Saturation Intensity)

A 24 bit image is often referred to as a full color image as it follows

$(2^8)^3 = 16,777,216$ Colors.



hue - the dominant color perceived by observe

Saturation - relative purity (or) how much white is m a pure color.

Brightness. Intensity is the same achromatic notation that we seen in grey level images.

HSI, Intensity & RGB

→ The hue is determined by an angle from a reference point usually red.