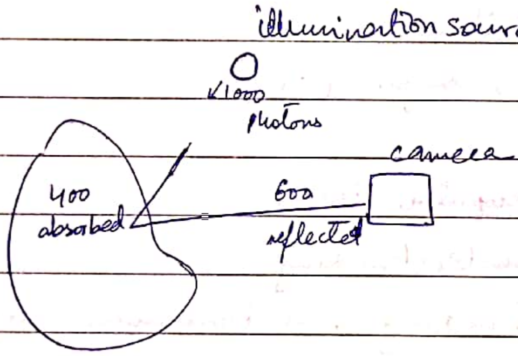
DIP by Rafael C. Gonzalez & Richard E. Woods 4th Ed.

Date

Digital Image Processing

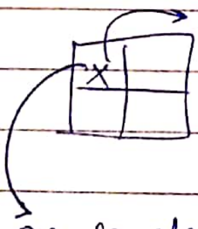
1. Point ops.
2. neighborhood based ops.
3. fourier theory
4. image restoration
5. image registration
6. morphological ops.
7. image segmentation
8. Pattern recognition

Digital Image: 2D image as finite set of digital values called pixels.



- pixel values represent gray levels, colors, heights, opacities
- digital img is approximation of real scene

- billions of photons fall on object
- camera if 1M Pixel, 1000 × 1000 pixels are captured



$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^n = n$

- if more colors in image then more image size
- for 8 colors, 3 bits
- for 4 colors, 2 bits

one sample per point: single value, black color or white color

$2^1 = 2$
0 or 1

0	1
1	1

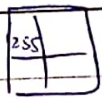
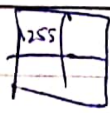
40	250
100	30

[0 ---- 255]

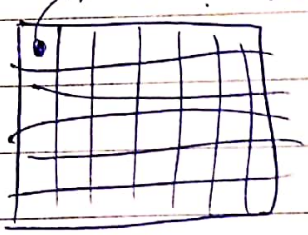
3 samples per point

white

Date

R 
G 
B 

0 ·· 255



1 MegaPixel image : each pixel takes one byte
so 1 MB

3 x 1 MP x 1 byte = 3 Mbytes

① Low level Process ← dip
- Input image , Output image
Example: noise removal, image sharpening

② Mid level Process ← dip
- Input Image, Output Attributes
Example: object recognition, segmentation

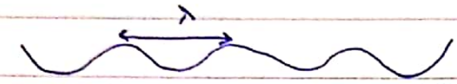
③ High level Process ← computer vision
- Input Image + Attributes , Output Understanding
Example : scene understanding , autonomous navigation

Uses of DIP : quality improvement, noise removal , contrast,
exposure / saturation

Date September 20th, 2021.

Chapter 2.

- every color is a different wavelength.



↓ diff. b/w 2 points on wavelength

★ Podium is of brown color. It reflects brown wavelength

- rods are for monochrometic light, cones are for colorful light

Human Visual System

1. Blindspot experiment
2. Brightness Adaptation & discrimination.
   - visual system undershoots or overshoots around boundary of region
3. Simultaneous Contrast
   - ~~[illegible]~~ blocks are of same intensity but due to varied background, percieved brightness changes.
4. Optical Illusions

Light + Electromagnetic Spectrum

- photon's speed: 3×10^8 m/s ← Speed is constant ~~less energy, more speed~~
- $f = \frac{c}{\lambda}$, $c = f\lambda$
- 380nm to 720nm ⇒ visible spectrum
- high frequency ⇒ high energy ⇒ high penetration power

Plank's constant = $6.62606957 \times 10^{-34}$

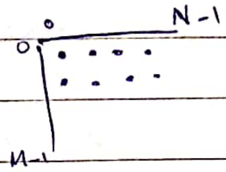
Date

Sampling, Quantisation + Resolution

(A) Image Representation

255 white
0 black

M × N picture
each dot represents pixel



1000 rows, 1000 cols = 1 MegPixel Camera

* $E = hf$ → freq.
↓
Planck's constant

• electrons released depend upon the type of photon hitting filter

• sensors have a sensing material, if digital camera it will have λ of visible light spectrum.

Chapter 02 DIP.

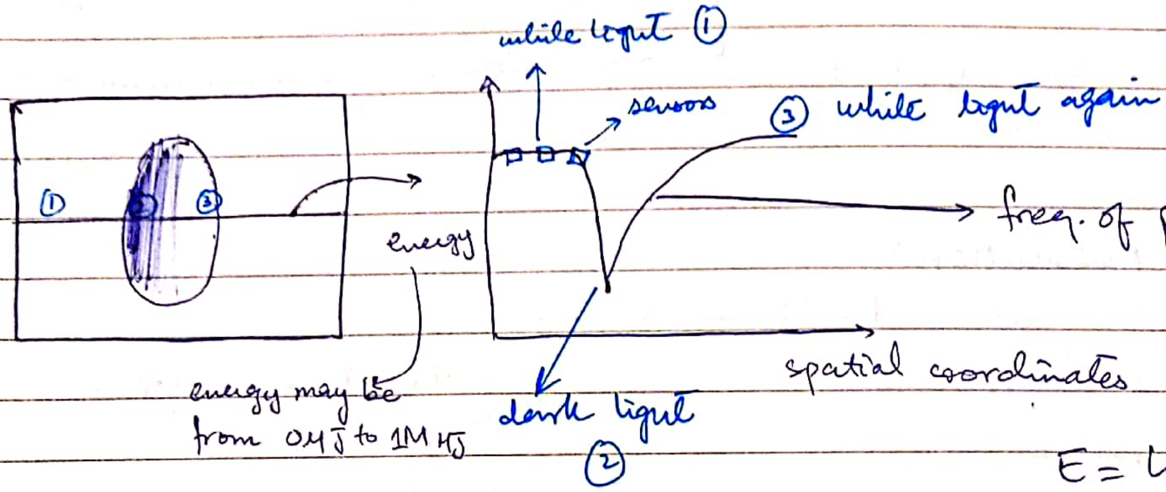
1. Sampling & Quantisation
2. Resolution — Spatial vs. Intensity
3. Basic Imaging Concepts
   i) Size and Dynamic Range
   ii) Brightness + Contrast
   iii) Image Histograms
   iv) Entropy
   v) Final note on Resolution

* eyes can distinguish 10 billion frequencies

Date

① Sampling + Quantisation

- images are 2D signals
- dark means no frequency

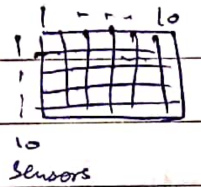


$E = h \times f$

energy ~~speed~~ depends on frequency of photon

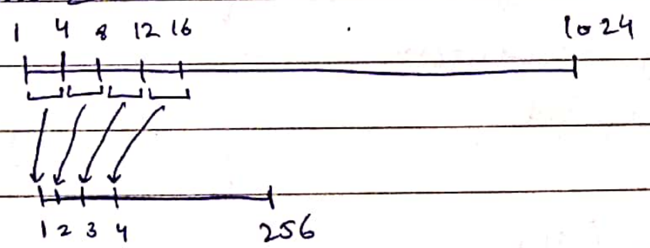
Sampling : Discretization of coordinate values is called Sampling

- every sensor captures ~~an~~ ~~image~~ coordinate values



10x10 — 100 MG Pixel camera

Quantization :

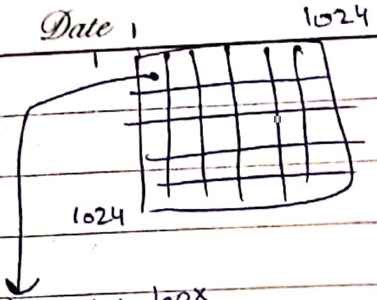


← we do it bcz we can only represent 256 colors in computers

Discretization of amplitude values of the sampled signal

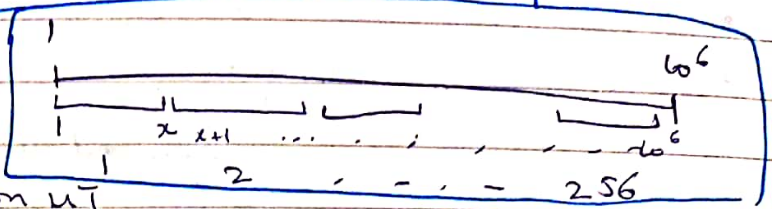
in sampling
- sensor capture values, quantization shows colors.
- we calculate energy levels of photons and decide how to represent colors at what values.

Date



Size of chunk = $\frac{1 \text{Million}}{256} = x$

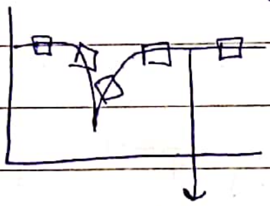
one way of Quantization



for every box energy can be b/w 0 μJ to 1 million μJ so we quantize it.

② Spatial Resolution:

- smallest discernable detail in an image
  ↓
  recognizable



pen : since there's no sensor here, pen will be visible but blur in image. increase no. of sensors

- sampling effects spatial resolution
  ↓ res. ↓ info

→ factors that matters in cameras
① sensors no.
② filter
③ lens
④ sensing material
⑤ internal geometry

Intensity Resolution: no. of intensity levels used to represent image.

~~↓ resolution, ↓ information — you loose information~~

- quantization effects Intensity Resolution
- 1 bit — 2 colors represented

intensity ↑ info ↑

2^1 = 0, 1 (2)
2^2 = 00, 01, 10, 11 (4)
2^3 = 000, 001, 010, 100 ... (8)
⋮
2^8 = 00000000 . . . 11111111 (256)

False Contouring: Prevalance of false borders in the images. You'll see correctly in 8 bits, do 7 bits you'll not see clearly and countour will be visible. This creates noise.

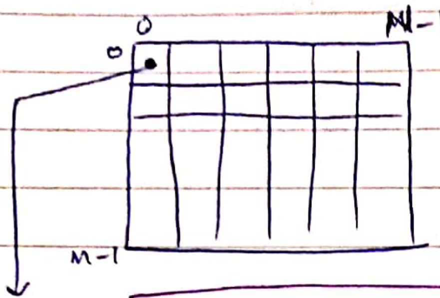
Date 28th September, 2021

(3) Basic Concepts

i) size ~~[illegible]~~ & dynamic range

↓

sampling + quantization contribute



M × N image

e.g. k bits → bit depth : no. of bits used to store 1 pixel of an image

size = M × N × K

$$g(x,y) = \begin{bmatrix} 0 & 40 \\ 200 & 255 \end{bmatrix} \text{8-bits}$$

size = 2 × 2 × 8 = 32 bits = 4 bytes

- cones can process 3 different wavelengths
- cuz they're in pair of 3
- 2D signal of image is represented by f(x,y)
  1D f(x), 3D f(x,y,z)
- PC monitors are 8-bits

* Dynamic Range: range of max + min values OR range of tonal difference b/w darkest + lightest light of image.

DR_1 = [0 – 0] (1) $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

DR_2 = [255 – 255] (2) $\begin{bmatrix} 255 & 255 \\ 255 & 255 \end{bmatrix}$

DR_3 = [0 – 255] (3) $\begin{bmatrix} 0 & 40 \\ 210 & 255 \end{bmatrix}$

- if low dynamic range, images are dull, washed out gray

* Contrast : difference b/w highest + lowest intensity

C_3 = 255 – 0 = 255

$[0 - 5]$ contrast $= 5 - 0 = 5$

Date

$$f(x,y) = \begin{array}{c|cccc|} & 0 & & & 1023 \\ 0 & 0 & 5 & 4 & \cdots & 5 \\ & 2 & 0 & 1 & \cdots & 4 \\ & \vdots & & & & \vdots \\ 1023 & 2 & 4 & 1 & \cdots & 5 \end{array}$$

→ if we change it to 255, contrast becomes $[255 - 0] = 255$

$1024 \times 1024 \times 8 \text{ bit} = 1\text{MB}$

(ii) Brightness & Contrast

* Brightness: average intensity values

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

$$B_3 = \frac{1}{2 \times 2} \times (0 + 40 + 200 + 255)$$

$$= \frac{495}{4}$$

$$= 123.75 \simeq 124$$

(better def.)

Contrast: average squared dist. from avg. intensity value

$$C_g = \sqrt{\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (f(i,j) - B)^2}$$

$$= \sqrt{\frac{1}{4} \times \left((0-124)^2 + (40-124)^2 + (200-124)^2 + (255-124)^2 \right)}$$

$= 107$. ← now even if one pixel has 255 value, the contrast doesn't shoot to 255 cuz this is avg.

* highest contrast is when half image is black + half is white.

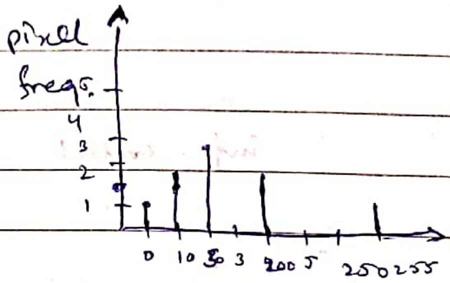
Date

iii) Image Histograms — distribution of pixel values

how to represent pixel values

i) pixel frequency based

$$f(x,y) = \begin{bmatrix} 200 & 255 & 50 \\ 50 & 10 & 200 \\ 0 & 50 & 10 \end{bmatrix}$$



factors that effect images

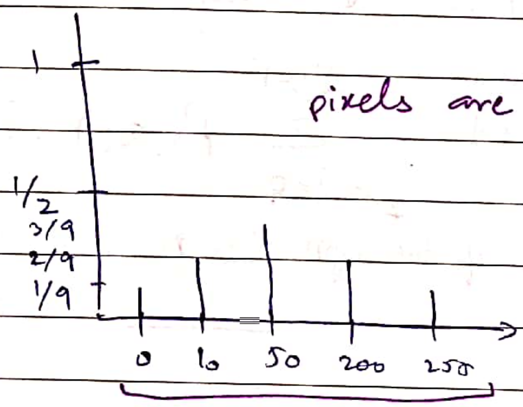
1 – photons vary

2 – environmental changes, waves

3 – cameras heat up; flow of current increases

flow↑ current, pixel values ↑

ii) prob. density function



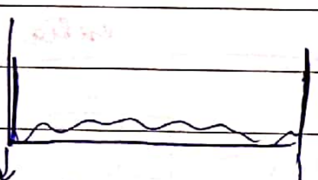
pixels are random variables

we can normalize these values b/w 0 and 1

0 0.1 0.2 0.3 . . . . 1.0

↓ darker side

↓ brighter side



↓ underflow
<0 values are collected at 0

↓ overflow

* good histogram is spread over all graph

* dynamic range of scene < dynamic range of detector = low contrast image (iv)

* incorrect exposure to light (iii)

dynamic range of scene + detector are matched

Saturation + Noise:

- Noise : any unwanted signal that exists in electronic systems

Saturation is a type of noise

↓

region with small variations in brightness.

Scanned with CamScanner

0 black
255 white

Date

(iv) Entropy :
- measure that tells about info of image
- it evaluates the degree of randomness of a system.

- repitition of msg decreases info of ~~msg~~ msg

info. content

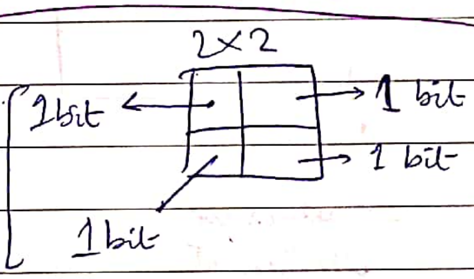
$$\underset{\text{information}}{I}\,(\underset{\text{event}}{E}) = \log_2\left(\frac{1}{P(E)}\right) \rightarrow \text{prob. of occurrence of event}$$

- Prob. ↓ Info ↑ or vice versa

entropy ← $H = \sum_{i=0}^{2^n-1} P(a_i) \cdot I(a_i)$ ← Range : $[0-n]$

if bit depth 8, $n = 256$

extra



$f(x,y)$	10	200	100
	10	100	20
	200	100	100

0	0	0
0	0	0
0	0	0

$H = P(a_{10}) \cdot I(a_{10}) + P(a_{20}) \cdot I(a_{20}) + P(a_{200}) I(a_{200})$

$= \frac{2}{9} \cdot \log_2\left(\frac{9}{2}\right) + \frac{1}{9}\log_2(9)$

$+ \frac{4}{9} \log_2\left(\frac{9}{4}\right) + \frac{2}{9}\log_2\left(\frac{9}{2}\right)$

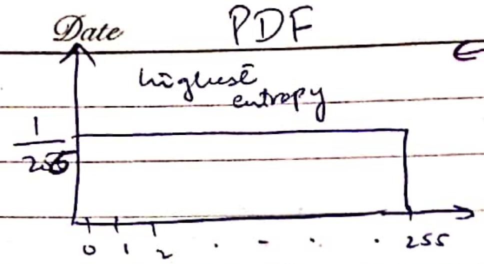
$=$

$\sum_{i=0}^{255} p(a_i)(I_i) = P(a_0) I(a_0)$
$+ P(a_2 0) I\, a_2 0 \ldots$

$= 1 \times 0 + 0 \times 0 \ldots$

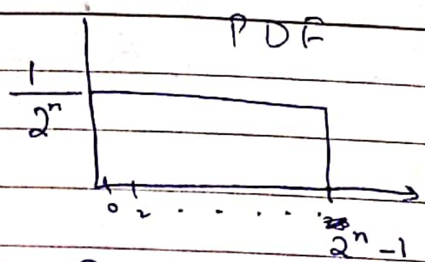
$= 0.$

Date



← maximum info

or



$$H = P(a_0) \cdot I(a_0) + P(a_1) \cdot I(a_1) + \dots + P(a_{2^n-1}) \cdot I(a_{2^n-1})$$

$$= \frac{1}{2^n} \log_2(2^n) + \frac{1}{2^n} \log_2 \cdot (2^n) + \frac{1}{2^n} \cdot \log_2(2^n) \dots$$

$$= \frac{1}{2^n} (n + n + n \dots n)$$

$$= \frac{1}{2^n} (2^n \cdot n) = n.$$

$\downarrow$

$n + n = 2n$

$n + n + n = 3n$

$n + \dots n^{2^n} = 2^n n$

Arithmetic Operations

- Arithematic operations on images are applied like in arrays.
- add, subtract, multiply, division

① Addition

$I_1 =$

250	60
40	200

$I_2 =$

50	200
200	40

$f = I_3 = I_1 + I_2 =$

300	260
240	240

→ overflow ↳ so normalize

image whose min. value may be 0 ← $f_m = f - \min(f)$

↓ modified image

$$f_R = K \cdot \frac{f_m}{\max(f_m)}$$

→ bit depth, k is the max value of desired bit depth

↓ resulting img

Date

$$f_m' = \begin{bmatrix} 60 & 20 \\ 0 & 0 \end{bmatrix}$$

$$f_R = 255 \times \begin{bmatrix} 1 & 1/3 \\ 0 & 0 \end{bmatrix} \Longrightarrow \begin{bmatrix} 255 & 85 \\ 0 & 0 \end{bmatrix}$$

5-bit 31
6-bit 63
8-bit 255

---

nature's values $\begin{bmatrix} 40 & \\ & \end{bmatrix}$

image 1 $\begin{bmatrix} 47 & \\ & \end{bmatrix}$ I_2 $\begin{bmatrix} 41 & \\ & \end{bmatrix}$

I_3 $\begin{bmatrix} 39 & \\ & \end{bmatrix}$ I_4 $\begin{bmatrix} 45 & \\ & \end{bmatrix}$

avg. values $= \dfrac{47+41+39+45}{4}$

$=$

- astronomy example
  many noise images added together reduce noise

Date October 4^{th}, 2021.

Basic Relationship of Pixels

1	1	6
0	7	7
0	1	7

3-bit image

- contrast ?
- entropy ?
- image modify to reduce entropy ?

• Neighborhood

$x \rightarrow$

x-1, y-1	x, y-1	x+1, y-1
x-1, y	x, y p	x+1, y
x-1, y+1	x, y+1	x+1, y+1

$y \downarrow$ 3×3

$q \in N_4(P)$

$N_4(P) = \begin{cases} x-1, y \\ x+1, y \\ x, y-1 \\ x, y+1 \end{cases}$

(4^{th} neighborhood)

↓

—, |,

$N_8(P) = \begin{cases} x-1, y \\ x+1, y \\ x, y-1 \\ x, y+1 \\ x-1, y-1 \\ x+1, y-1 \\ x-1, y+1 \\ x+1, y+1 \end{cases}$

8^{th} neighborhood

\ / | —

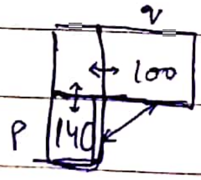
• Connectivity

→ adapted from neighborhood relation

- 2 pixels are connected if share same class i.e. shades of same color OR same range of intensity

- 4^{th} connectivity
- 8^{th} connectivity
- mixed connectivity : p and q are m-connected if

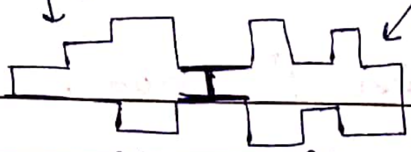
$$q \in N_4(P) \text{ or } \underbrace{q \in N_D(P)}_{\text{diagonal neighbor}} \text{ and } N_4(P) \cap N_4(q) = \underbrace{\emptyset}_{\text{not belonging to same class}}$$



example

class range [100 - 140]

Date

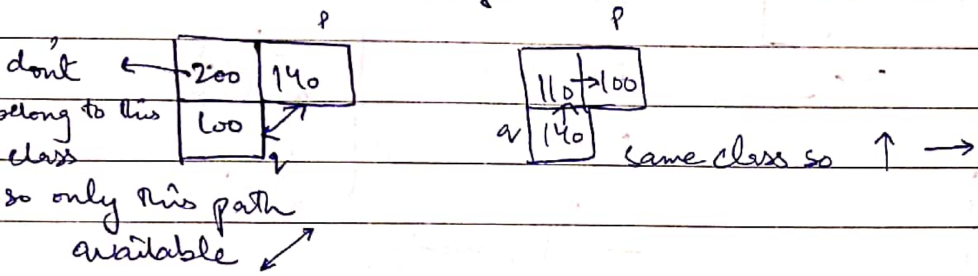


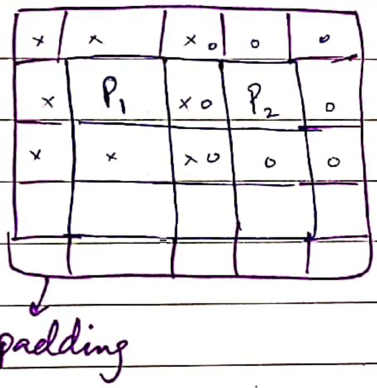
- Adjacency :

p is adjacent to q if they are 'connected'.

adjacent if atleast one pixel connected of both together.

- Path   class range [100 – 140]





as P changes, neighbors change

- Distance :

$$p, q, z \quad (x,y) \; (s,t) \; (u,v)$$

$D_e(p,q)$ = Euclidean distance = $\sqrt{(x-s)^2 + (y-t)^2}$

D is a distance fn. if

1. $D(p,q) \geqslant 0$
2. $D(p,q) = D(q,p)$.
3. $D(p,z) = D(p,q) + D(q,z)$

Date

Chapter 3

Image Enhancement

⟹ Objectives to do enhancement:

i) improvement of overall picture info.
[dark image → bright image
low contrast → high contrast]

ii) removal of noise from images

iii) to highlight part of an image

• pt. processing:
processing of each pixel

⟹ Enhancement operation types:

i) Spatial (pixel) values based ops. (applied directly on pixels)

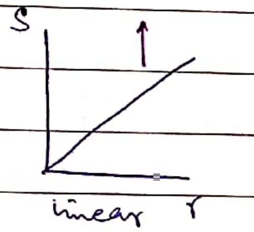
ii) Frequency values based ops.

i) Basic spatial domain image enhancement

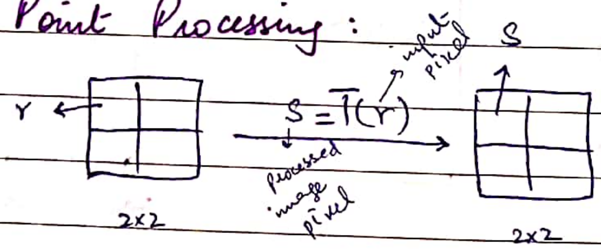
$$g(x,y) = T[f(x,y)]$$

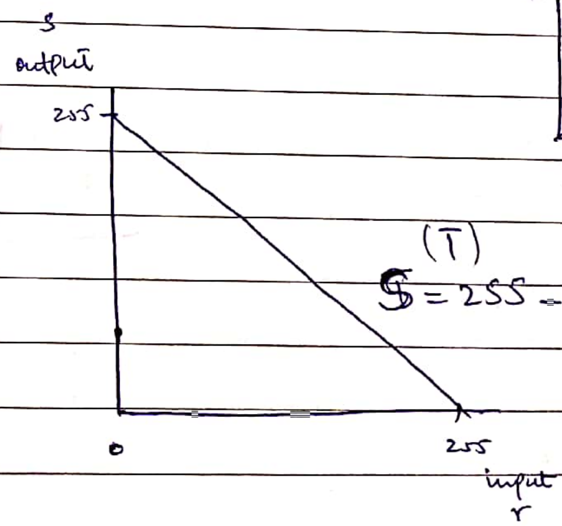
$g(x,y)$: processed image
T: operator / transformation / function
$f(x,y)$: input

doing nothing
$s = r$



Point Processing:





• negative image:
intensity values are inverted

Example: Thresholding

useful for segmentation when we want to isolate an object of interest from background.

$$s = \begin{cases} 255 ; & r \geq T \\ 0 ; & r < T \end{cases}$$

← for normalized image

code log. eq. using opencv

Date

does nothing 1 ↑ | more dark, less light ↑ 2 inverts | 3

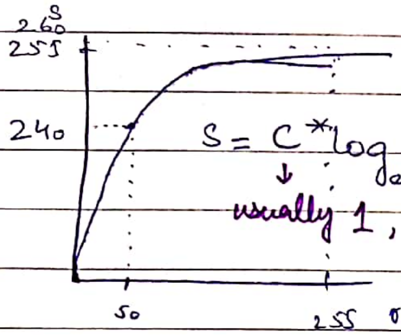
Point Operations: Linear, Negative image, Thresholding, Log Transformation

$s = 255 - r$ (if 8 bits)

or

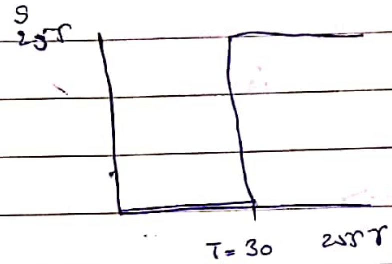
$s = (2^n - 1) - r$ (if n bits)

- simplest type of image segmentation.
- works if different intensities of fg + bg.



$s = c * \log_e (1 + r)$

c → usually 1, scaling constant



- if image has lot of dark regions, log will brighten it.
- inverse log transformation performs opposite transformation
- log function will map darker ↓values regions to lighter regions ↑values

$f(x,y) = \begin{bmatrix} 10 & 5 \\ 20 & 15 \end{bmatrix}$

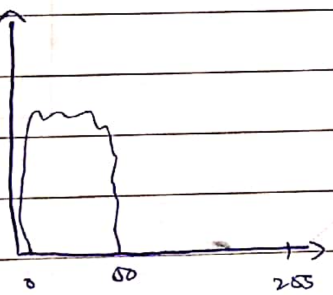
$f(0,0) = c * \log_e(1+10)$

$f(0,1) = c * \log_e(1+5)$

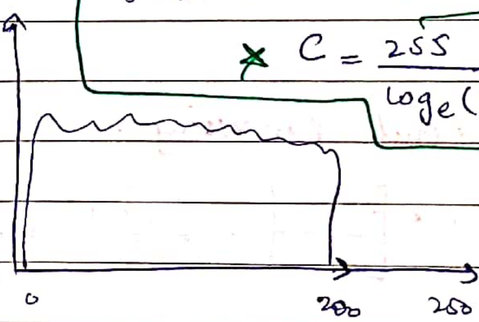
$f(1,0) = \quad (1+20)$

$f(1,1) = \quad (1+15)$

$c = \frac{255}{\log_e(20)}$ → as 8-bit



$s = \log_e(1+r)$ ⟶



$f(x,y) = \begin{bmatrix} 10 & 5 \\ 20 & 15 \end{bmatrix} \longrightarrow g(x,y) = \begin{bmatrix} 130 & 70 \\ 255 & 205 \end{bmatrix}$

write algo of Log Transformations

Assignment

10	5
20	15

apply Power Law Transformation

Date October 7th, 2021.

Point Ops. : $s = T(r)$

① Linear $s = r$.

② Image Negative $s = L - r$

③ Threshold transformation $s = \begin{cases} r < T; & 0 \\ r \geq T; & 255 \end{cases}$

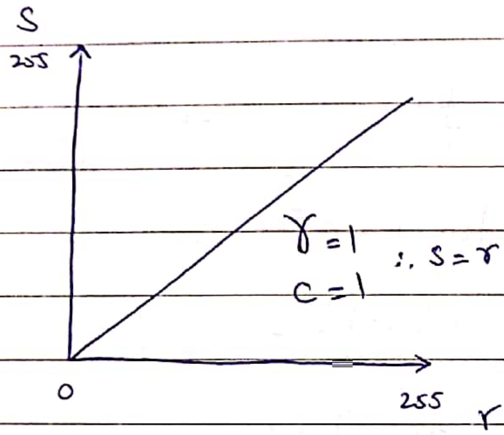
④ log transformation

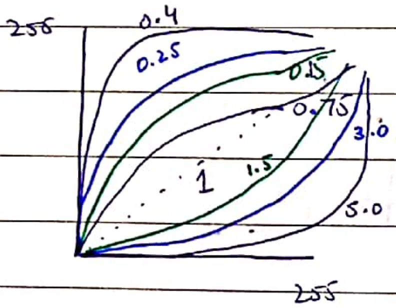
$$s = c * \log_e(1+r)$$

+1 cuz undefined on 0.

⑤ Power Law trans.

$$s = c * r^{\gamma}$$ $\gamma \to$ gamma





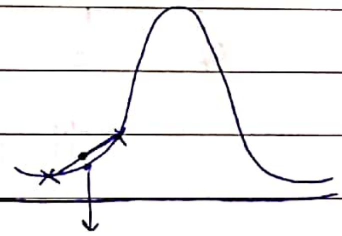
$\gamma < 1 \downarrow$ brightness $\uparrow$

$\gamma > 1 \uparrow$ brightness $\downarrow$

Difference from Log trans.

* PLT gives more freedom, pick γ of your choice

* 4K image takes too much time? downsample using good algo (reduce pixel resolution)



find this, no sensor

X sensors

- is given by HW
- is given by SW

so by HW is better

night

0	0
0	0

bilinear

NN

bicubic

interpolation types

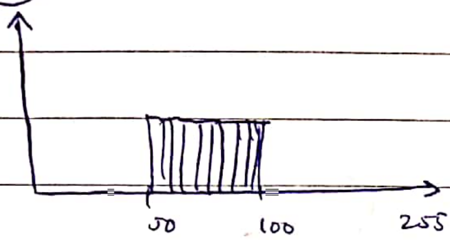
$$s = c * \log_e(1+0)$$

* here log transformation can't do anything

Date

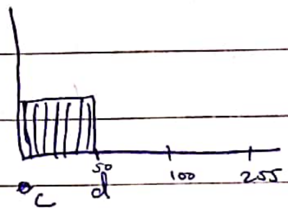
(6) Contrast Stretching Transf.

$$f(x,y) = \begin{array}{|c|c|} \hline 80 & 50 \\ \hline 100 & 90 \\ \hline \end{array}$$



histogram

$$s = (r-c)$$
$$= 50 - c \rightarrow$$



Contrast Stretching

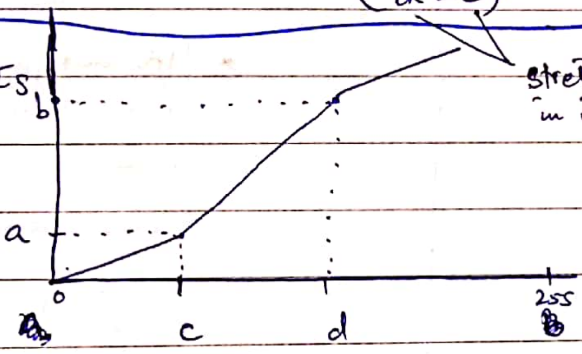
upper + lower intensity limit

$$\boxed{s = (r-c) \times \frac{(b-a)}{(d-c)} + a}$$

stretching points in input image

30	0
50	40

80−50 → 30; 50−50 → 0; 100−50 → 50; 90−50 → 40



mapping small range into large range

$$f(0,0) = (80-50) \times \frac{255-0}{100-50} + 0$$
$$= 153$$

$$f(0,1) = 0 \times \frac{255}{50} = 0$$

$$f(1,0) = 50 \times \frac{255}{50} = 255$$

$$f(1,1) = 40 \times \frac{255}{50} = 204$$

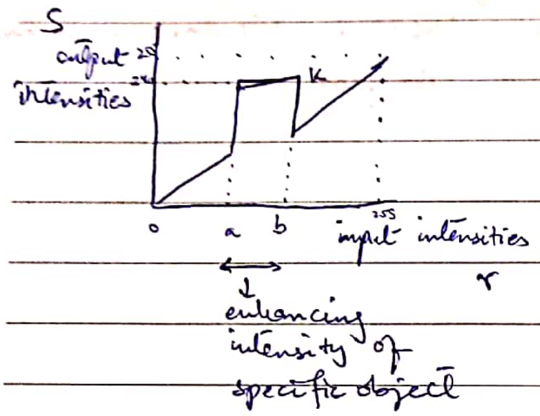
153	0
255	204

Scanned with CamScanner

Date October 14th, 2021.

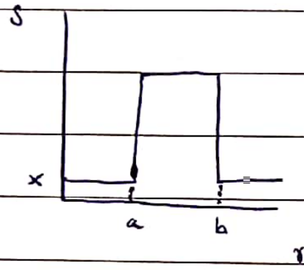
⑦ Image (intensity) Slicing / Gray level Slicing
⑧ Bit-plane slicing
⑨ Histogram processing

Image (Intensity) Slicing : used to highlight some parts of the image
or Gray level slicing



$$s = T(r) = \begin{cases} a \le r \le b \;;\; k \\ \text{otherwise} \;;\; r \end{cases}$$

if in this range then enhance I to K



$$s = T(r) = \begin{cases} a \le r \le b \;;\; k \\ \text{otherwise} \;;\; x \end{cases}$$

Bit plane slicing: to partition an image into individual bit planes

$$f(x,y) = \begin{array}{|c|c|} \hline 2 & 3 \\ \hline 4 & 255 \\ \hline \end{array}$$ 8 bit image

	MSB -8 -7 -6 -5	LSB -4 -3 -2 -1
2 =	0 0 0 0	0 0 1 0
3 =	0 0 0 0	0 0 1 1
4 =	0 0 0 0	0 1 0 0
255 =	1 ϕ ϕ ϕ	ϕ ϕ ϕ ϕ

← more info

-8 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 0 & \phi \\ \hline \end{array}$ -7 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 0 & \phi \\ \hline \end{array}$ -6 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 0 & \phi \\ \hline \end{array}$ -5 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 0 & \phi \\ \hline \end{array}$

-4 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 0 & \phi \\ \hline \end{array}$ -3 $\begin{array}{|c|c|} \hline 0 & 0 \\ \hline 1 & \phi \\ \hline \end{array}$ -2 $\begin{array}{|c|c|} \hline 1 & 1 \\ \hline 0 & \phi \\ \hline \end{array}$ -1 $\begin{array}{|c|c|} \hline 0 & 1 \\ \hline 0 & \phi \\ \hline \end{array}$

→ less info

• used in image compression
image transformation, transfe

Size : $M \times N \times k$ bits

1M × 8-bits = 8 MPixel

Date gray scale images

1M × 2-bits = 2 MPixels

(Dollar bill example)

- bg dark, fg light = information

130	255
253	129

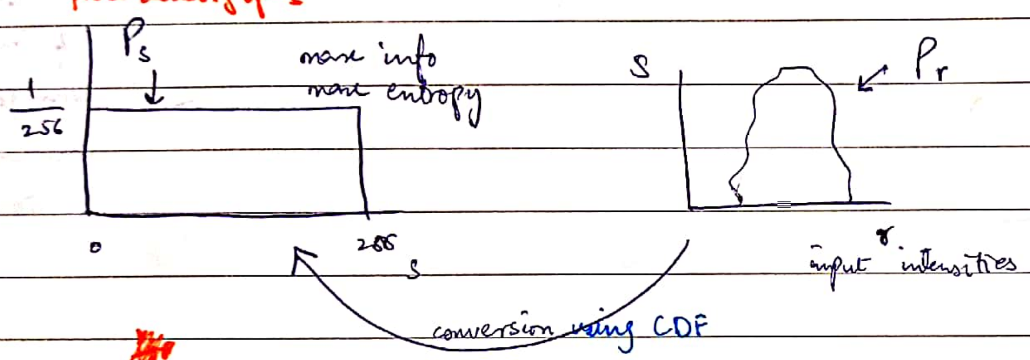
	128	64	32	16	8	4	2	1
130 =	1	0	0	0	0	0	1	0
255 =	1	1	1	1	1	1	1	1
253 =	1	1	1	1	1	1	0	1
129 =	1	0	0	0	0	0	0	1

discard → if keep only 8th bit, less info, include 7th, more info, add 6th more info and so on

3 =	0	0	0	0	0	0	1	1

discard → in this case, if we discard last 6, 3 is dark, 0 is dark no problem

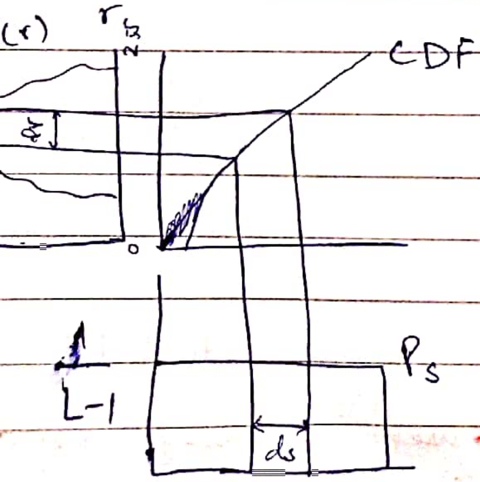
Histogram Processing:



$T(r)$ = CDF (Commulative Density Fn.)

$$s = T(r) = (L-1)\int_0^r P(r)\,dr$$

no. of colors you want



equalisation of $P(r)$ to P_s using CDF of analogue images

$$P_r \cdot dr = P_s \cdot ds$$

$$P_s = \frac{P_r \cdot dr}{ds}$$

slope of r w.r.t s

$$P_s = P_r \cdot \frac{1}{\frac{ds}{dr}} \quad \text{(A)}$$

(solved next pg)

$$P_s = P_r \cdot \frac{1}{(L-1)\cdot P_r}$$

$$P_s = \frac{1}{L-1}$$

Date

$$\frac{ds}{dr} = (L-1) \frac{d}{dr} \int_0^r P(r)\, dr$$

$$\frac{ds}{dr} = (L-1) \cdot P(r)$$

- why we do equalisation?

contrast ↑

entropy MAY get better