Computer Graphics and Image Processing
CSE 455

Computer Graphics and Image Processing Laboratory
CSE 456

Caution:

Sudden CT

1st July, 2025 – 1st Class

02.07.25 – 2nd Class

Image processing:

Image information:

signal (main component) → frequency, amplitude

image $\xrightarrow{\text{is transformed}}$ signal

Cybersecurity

Spatial domain → amplitude

Why need image transformation?

Application:

Theory

DFT — Discrete Fourier Transformation

DCT — (math)" Cosine "

DFT, DCT difference:

- DCT → real part
- DFT → real and complent part

DCT performs better image compression than DFT.

$$C(u) = \alpha(u)\sum_{x=0}^{n-1} f(x)\cos\left(\frac{(2x+1)u\pi}{2N}\right)$$

$$\alpha(K) = \begin{cases} \sqrt{\frac{1}{N}} & \text{if } k=0 \\ \sqrt{\frac{2}{N}}; & \text{otherwise} \end{cases}$$

Homework:

$C(1), C(2) \ldots C(7)$?

$$C(0) = \sqrt{\frac{1}{8}} \sum_{x=0}^{7} f(x) \cos\left\{\frac{(2x+1) \cdot 0 \cdot \pi}{2N}\right\}$$

$$= \sqrt{\frac{1}{8}} \sum_{x=0}^{7} f(x) \cos 0°$$

$$= \sqrt{\frac{1}{8}} \left\{f(0) + f(1) + f(3) + \ldots + f(7)\right\}$$

$$= \sqrt{\frac{1}{8}} \left\{20 + 12 + 18 + 56 + 83 + 10 + 109 + 119\right\}$$

DDA (Digital

(Line Drawing algorithm)

Math:

1. Enter two endpoints. (x_1, y_1), (x_2, y_2) — $m < 1$
2. Calculate, $m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1}$ — $m > 1$

while (endpoints != endpoints) { — $m = 1$

3. if $(m < 1)$

{
$x_{k+1} = x_k + 1$
$y_{k+1} = y_k + m$
$y' = \text{roundup}(y_{k+1})$
print (x_{k+1}, y')
}

else if $(m > 1)$

{ $y_{k+1} = y_k + 1$
$x_{k+1} = x_k + \frac{1}{m}$
$x' = \text{roundup}(x_{k+1})$
print (x', y_{k+1})
}

else $(m = 1)$ { $x{+}{+}$ $y{+}{+}$ } print (x, y) }

Math (DDA)

$(2,3), (12,8)$ $m = \frac{8-3}{12-2} = \frac{5}{10} = 0.5$

$(2,3)$
$(3,3.5) \longrightarrow (3,4)$
$(4,4) \longrightarrow (4,4)$
$(5,4.5) \longrightarrow (5,5)$
$(6,5) \longrightarrow (6,5)$
$(7,5.5) \longrightarrow (7,6)$
$(8,6) \longrightarrow (8,6)$
$(9,6.5) \longrightarrow (9,7)$
$(10,7) \longrightarrow (10,7)$
$(11,7.5) \longrightarrow (11,8)$
$(12,8) \longrightarrow (12,8)$

Disadvantage:

→ Simple, not efficient
→ fraction, takes much time

Then came,
Bresenham line algorithm.

* Advantages of DCT.
* Hadamard transform. (important for exam)
Hadamard transform example. A ⟷ B
" " advantage

Digital Image Processing:

Steps:

i) Problem domain
ii) Image acquisition (using camera, scanner)
iii) Image enhancement. (by modifying / editing)
iv) Image restoration
v) Color image processing
vi) Wavelet and Multi-resolution processing.
vii) Image compression → lossy, → lossless
viii) Morphological Image Processing.
ix) Image segmentation.
x) Image representation and Description.
xi) Object recognition

Components of Image processing System:

i) Image sensors.
ii) Specialized Image Processing Hardware.
iii) Computer

⇒ What is color?

⇒ Physical properties of light

⇒ Color model
- → RGB
- → CMY
- → HSI

⇒ Conversion from RGB to HSI
(Equation দেখতে হবে)

⇒ Relationship between pixels.

Image Processing Idea:

Image Compression:

⇒ Huffman code

Original Source → Source reduction

** Exercise.

Arithmetic Coding → Arithmetic encoding
→ Arithmetic decoding.

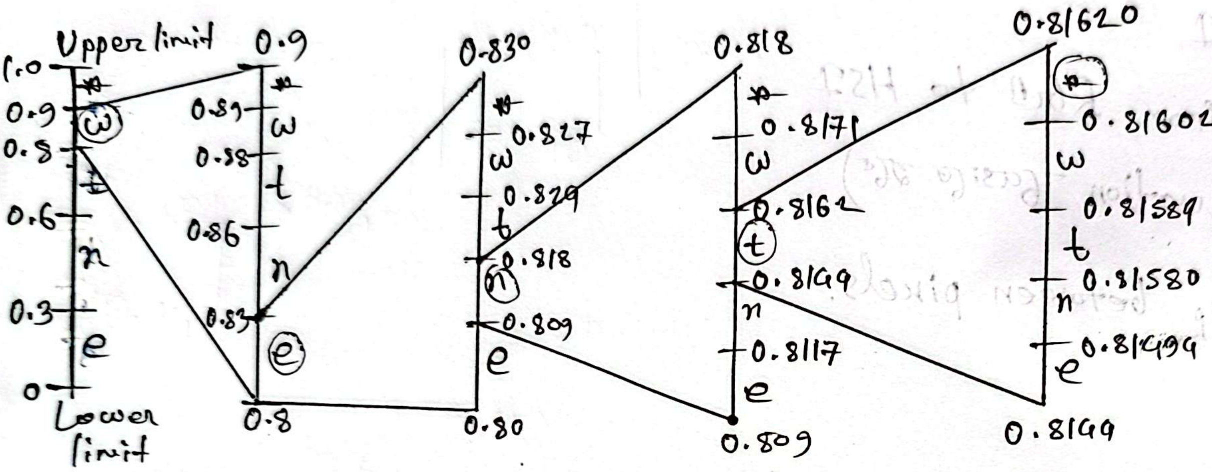
** Arithmetic Coding – Example

⇒ Bit plane slicing

Arithmetic encoding:

Q. Encode the msg "Went*" — find codeword

Symbols :	e	n	t	w	*
Probabibility :	0.3	0.3	0.2	0.1	0.1



Range of symbols = Lower limit + d (probability of symbols)

d = Upper limit − Lower limit

Final range:

0.81602 − 0.81620 $\}\rightarrow$ codeword [illegible]

Hence the arithmetic codeword is 0.81602 — 0.81620

$$\text{Generation tag} = \frac{\text{Upper limit of codeword} + \text{Lower limit of codeword}}{2}$$

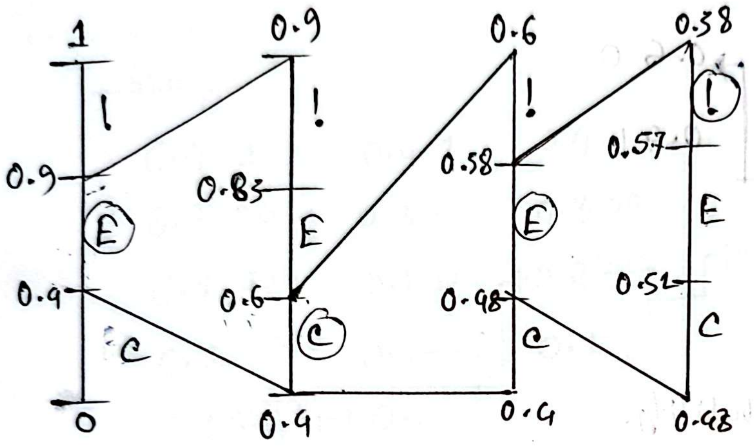
$$= \frac{0.81602 + 0.81620}{2}$$

$$= 0.81611$$

Arithmetic decoding:

Decode message 0.572, given the coding model
$p(C) = 0.4$, $p(E) = 0.5$, $p(!) = 0.1$

Ans:

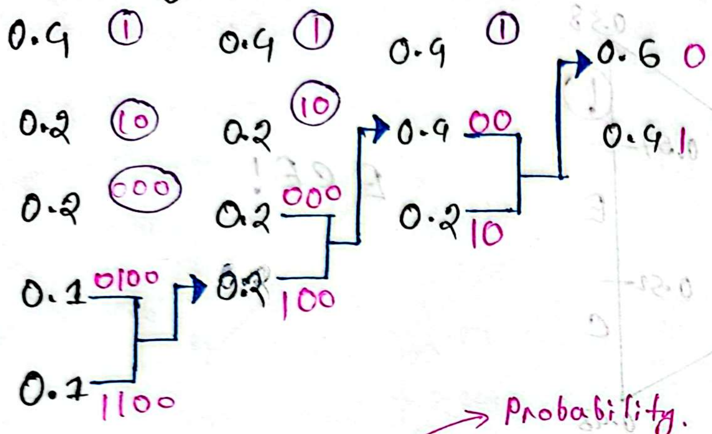


ECE!

Huffman Coding:

Symbol $\rightarrow$ S_0 S_1 S_2 S_3 S_4
Probability $\rightarrow$ 0.4 0.2 0.2 0.1 0.1

} Sort them in decreasing order if they are not.



Probability.

$$\text{Length}, L_i = \sum_{i=1}^{n} P_i(L)$$

$$= (0.4\times1) + (0.2\times2) + (0.2\times3) + (0.1\times4) + (0.1\times4)$$

$$= 2.2 \text{ bits/pixel.}$$

$$\underset{(\mu)}{\text{Entropy}} = \sum_{i=1}^{m} P_i \log_2\left(\frac{1}{P_i}\right)$$

$$= 0.4\times\log_2\left(\frac{1}{0.4}\right) + \cdots\cdots + 0.1\times\log_2\left(\frac{1}{0.1}\right)$$

$$= 2.121$$

$$\text{Efficiency}, \eta = \frac{\mu}{L_i}$$

$$= \frac{2.121}{2.2}$$

$$= 0.96$$

$$\text{Redundancy, } (\gamma) = 1 - \eta$$
$$= 1 - 0.96$$
$$= 0.04$$

Symbol	Probability				
	0.4 1	0.4 1	0.4 1	0.4 1	0.6 0
	0.3 00	0.3 00	0.3 00	0.3 00	0.4 1
	0.1 110	0.1 110	0.2 010	0.3 10	
	0.1 0010	0.1 0010	0.1 110		
	0.06 01010	0.1 1010			
	0.04 11010				

$$L_{avg} = (0.4 \times 1) + (0.3 \times 2) + (0.1 \times 3) + (0.1 \times 4) + (0.06 \times 5) + (0.04 \times 5)$$
$$= 2.2 \text{ bit/pixel} \approx 3 \text{ bit/pixel}.$$

Symbol	Probability				
B	0.30 1	0.30 1	0.30 1	0.45 1	0.55 0
F	0.25 00	0.25 00	0.25 00	0.30 00	0.45 1
A	0.20 11	0.20 11	0.25 01	0.25 10	
D	0.15 001	0.15 001	0.20 11		
C	0.06 0101	0.10 101			
E	0.04 1101				

$$L_{avg} = (0.30 \times 1) + (0.25 \times 2) + (0.20 \times 2) + (0.15 \times 3) + (0.06 \times 4) + (0.04 \times 4)$$
$$= 2.05 \text{ bits/pixel}.$$

DCT → Discrete Cosine Transformation:

CGIPLab (report guideline)

Front page
Report title
Objective
Necessary tools
Description
Algorithm
Pseudocode
Activity diagram
Source code
Output description (Landscape print)
Result explanation (Calculation)
Limitation of code
~~Conclus~~ Problem faced.
Conclusion.
Reference. (IEEE format)

1

DDA

1. Choice (Cmy-HSI / HSI-CMY / RGB-HSI / RGB-CMY)
2. You have chosen this one ....
3. Enter the CMY values
4. HSI will be [ ]

DDA line dra
↳ expensive (floating point, round operation costs much)

Deriving
Bresenham line algorithm:

Decision parameter — (p) — decides upper on lower pixel.

P → minus < 0
x → x+1
y → unchanged

$$P_{k+1} = P_k + 2\Delta y.$$

else,
x → x+1
y → y+1

$$P_{k+1} = P_k + 2\Delta y - 2\Delta x$$

two endpoints,

(20,10) (30,18)

$P_0 = 2\Delta y - \Delta x$

$= 2(18-10) - (30-20)$

$= 2 \times 8 - 10$

$= 6$

(20, 10)

$P_0 = 6 \rightarrow (21, 11)$

$P_1 = 2 \rightarrow (22, 12)$

$P_2 = -2 \rightarrow (23, 12)$

$P_3 = 14 \rightarrow (24, 13)$

$P_4 = 10 \rightarrow (25, 14)$

(26, 15)

(27, 16)

(28, 16)

(29, 17)

(30, 18)

Circle drawing algorithm:

$x^2 + y^2 = r^2$

circle — 8 way symmetric.

ellipse — 4 way symmetric

* Eight-way symmetry: (~~neesity~~ necessity — requires less computation time)

* Mid-point circle algorithm.

Initial decision parameter — P_0

Circle Plot starting → (0, r)

Question : notice quadrant if there

CGIP - Graphics: 2.4

* Midpoint ~~A~~ Ellipse Algorithm — স্যার বাদ দিয়েছেন

* Polygon filling:

i) Boundary fill algorithm   ii)

* C-Curve: break in 90°

* Koch-curve:

break middle part in 60°

* The Sierpinsky Gasket:

* Area filling algorithm.

→ Boundary fill → uses stack.

→ 4-connected → problem (can't color fully some structure)

→ 8-connected.

solves this problem.

→ Flood-fill (color diff. whereas same in boundary fill)

* Aliasing and Anti-aliasing problem:

(pixel-breaking -ভাঙ্গায়)

* Unequal brightness এর জন্য aliasing problem হতে পারে।

CamScanner

→ Unweighted filtering

→ Super-sampling.

→ Character display.
- → Bitmap font
- → Outline font.

Picket fence problem: (definition)

	বাকিটার
1, 2, 3 9, 11	4, 5, 6, 8

Chapter 4, 5. (two dimension)
6, 8 (three dimension)

Chapter—Four:

⇒ What is transformation?

⇒ Translation.
figure: 4.2

Math
4.9
4.3
97 page

⇒ Rotation.

⇒ Scalling.

⇒ Mirror reflection.

4.2 Composite transformation
(sum up of previous points)

⇒ Instance transformation.