Image Compression -

- The same information can be represented by different amount of data..
- In compression
  * reduce the amount of data required to represent a given quantity of information while preserving as much information as posible.

Data compression - refers to the process of reducing the amount of data required to represent a given quantity of information.

Data is the means by which information is conveyed.

⇒ Various amount of data can be used to represent same amount of information.

⇒ Representation that contain irrelevant or repeated information are said to contain redundant data.

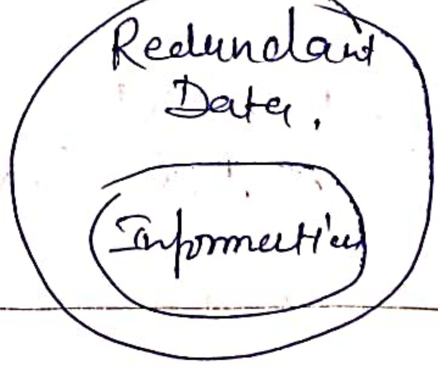


Image compression - It is the art & science of reducing the amount of data required to represent an image.

- The No. of images that are compressed and decompressed daily is staggering and the compressions and decompressions themselves are virtually invisible to the user.

- Anyone who ~~owns~~ a digital camera, surfs the web, or watches the latest hollywood movies (on dvds or yify torrents) benefits from the compression algorithms.

In terms of image, information corresponding to the object captured in the image, their shape, their color etc.

The different ways to represent this information can be either

→ Through pixel intensities (spatial domain (basic) or

→ Through extracting features of that image (higher level of information eg - edge image. Edge out of prev. image) or

- Through simply noting the frequencies of the intensity values existing in the image along with their location (frequency domain), etc.

Dataset1 $\xrightarrow{\text{Compression}}$ Dataset2

image — compressed image

n_1 carrying units (eg bits) — n_2 carrying ~~bits~~ units (eg - bits)

Two representations of same information

Compression Ratio

$$C = \frac{n_1}{n_2}$$

Relative Data redundancy of the representation with n_1 Bits.

$$\boxed{R = 1 - \frac{1}{C}}$$

- Image compression involves reducing the size of image data files, while retaining necessary information

- Retaining necessary information depends upon the application.

- The reduced file created by the compressed file and is used to reconstruct the image resulting in the decompressed image.

- The original image, before any compression is performed is called the uncompressed image file.

- The ratio of the original, uncompressed image file and compressed file is referred to as the compression ratio.

Q-1 ABC wish to downloaded the XYZ movie in 720P (1280X720) format. The movie is of 2 hours duration and was recorded at 60 fps (frames per second)

1) Calculate the amount of Storage it would require to store this movie?

2) How many dual layer blue-ray disk would be required to store this movie??

3) How much time would it take to transmit this video over broadband connection having bandwidth capacity of 100 MbPS??

720P → $\frac{16}{9} = \frac{x}{720}$ P → 16:9 $x = \frac{720 \times 16}{9} = 1280$

Ans.

Q-1 A video is a seq. of video frames where each frame is a full color still image.

1) Storage requirement of 1 sec of video

1 frame is one RGB image → 3 Bytes

video frame → 1280 × 720 × 3 Bytes

60 FPS × (1280 × 720) × 3B ≈ 158.2 MB
PPF
(Pixels Per frame)

2) storage requirement of complete 2 Hr. movie

158.2 × 60 × 60 × 2 ≈ 1113 GB

No. of blue-ray (having capacity of 50GB) required = 1113 / 50 = 23

3) Time required to transmit complete 2 hours movie:

100 MbPS → 100 Mb in 1 sec

$100\ MbPS \rightarrow 100\ Mb$ in 1 sec

$100 \times 1024 \times 1024$ bits $\rightarrow$ 1 sec

$1113 \times 1024 \times 1024 \times 1024 \times 8$ bits

$$= \frac{1113 \times \cancel{1024} \times \cancel{1024} \times 1024 \times 8}{100 \times \cancel{1024} \times \cancel{1024}}$$

$\simeq 25.33$ hours.

This is the Power & need of image Compression

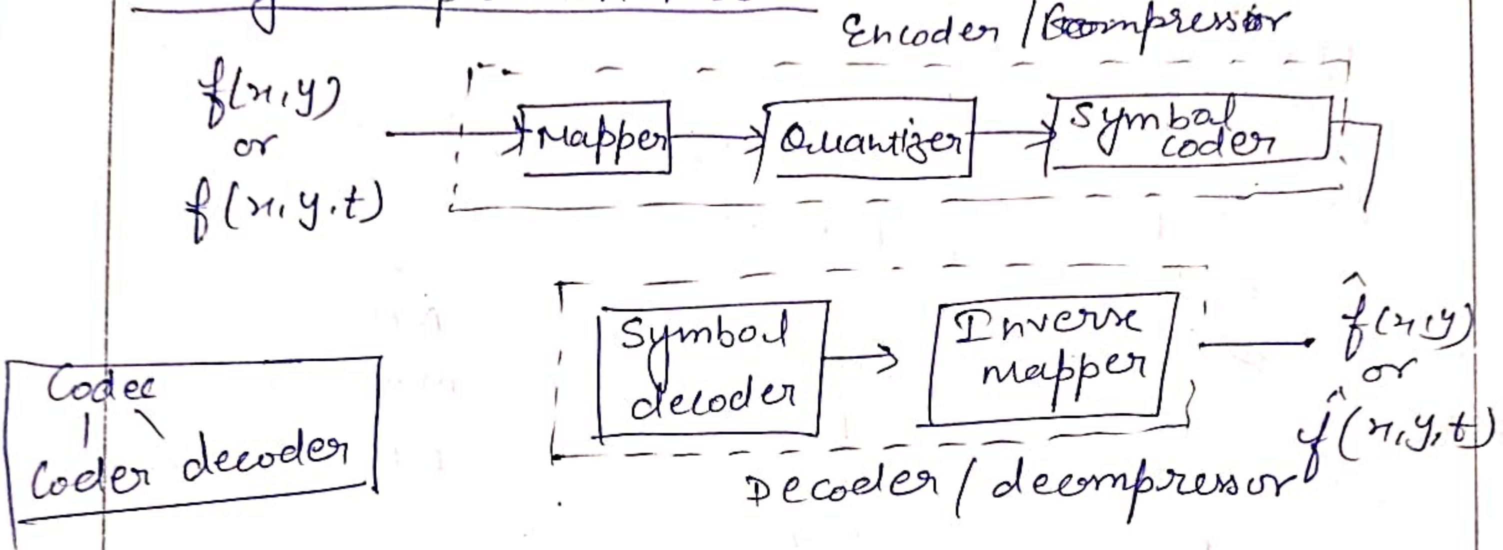
Applications

- web page image
- high resolution digital camera photos
- Tele-video conferencing
- Remote sensing
- Document & medical imaging
- Fax transmission.

\# unsplash
\# Pixa

] good quality image as compare to google image.

Image compression Model



functional block diagram of a general image compression system.

Image Formats - are used/ ~~in~~ standardized means of organizing and storing digital images.

Image container - is one that handles multiple types of image data

Image compression standards defines the rules or procedures for compressing and decompressing images.

Image compression Standards, formats & Containers

- Still Image
  - Binary
    - CCITT (Group3)
    - CCITT (Group4)
    - JBIG (or JBIG 1)
    - JBIG2
    - TIFF
  - Continious tone
    - JPEG
    - JPEG-LS (near-lossless)
    - JPEG-2000
    - BMP
    - GIF
    - PDF
    - PNG
    - TIFF
- Video
  - DV
  - H.261
  - H.262
  - H.263
  - H.264
  - MPEG-1
  - MPEG-2
  - MPEG-4
  - MPEG-4 AVC
  - AVS
  - HDV
  - M-JPEG
  - Quick Time
  - VC-1 (or WMV9)

Standard	Applications
JPEG	Still image compression (variable Bit rate)
H.261	Video Conferencing over ISDN
MPEG-1	Video on digital Storage Media
MPEG-2	Digital Television
H.263	Video telephony over PSTN
MPEG-4	object based coding, synthetic content
JPEG-2000	Improved Still image compression
H.264/MPEG4 AVC	Improved video compression.

Compression Algorithm and its Types -

is used to reduce the source data to a compressed form and decompress it to get the original data.

Any compression Algo has 2 components -

Modeller

- It is used to ~~contain~~ condition the image data for compression using the knowledge of data
- It is present in both sender and receiver
- It can be either static or dynamic

Coder

- Sender side Coder is known as encoder
- This codes the symbols independently or using the model
- Receiver side Coder is known as decoder
- Decoder decodes the message from the compressed data

Compression Algo...

- Lossless compression.
- Reconstructed data is identical to the original data (Entropy coding)
- ~~Ex~~ techniques.
  Huffman Coding
  Arithmetic Coding
  Shanon-Fano coding
- eg - Legal and Medical documents.

Lossy compression
- Reconstructed da... is an approximati... to the original data (source coding)
- eg
  - Linear Predictio...
  - Transform Coding

Differences

Lossless	Lossy
1) Reversible Process	1) Non-reversible process
2) No information is lost	2) Some information is lost
3) Compression Ratio is less	3) Compression Ratio is very high
4) It is used for that data which human can handle directly like Text data	4) It is used for diffused data which human can not understand or interpret.
5) Compression is independent on Psychovisual System	5) compression is dependent on Psychovisual system

Another way of classifying image Compression Algorithm

1)- Entropy coding - The average information in an image is known as its entropy.

- Coding is based on the entropy of the source and on the possibility of occurance of symbols.

- An event that is less likely to occur is said to contain more information than an event that is more likely to occur.

- set of symbol (alphabet)

$$S = \{S_1, S_2, \ldots S_N\}$$

N is No. of symbols in the alphabet.

- Probability distribution of the symbols

$$P = \{P_1, P_2, \ldots P_N\}$$

According to shannon, the entropy H of an information source S is defined as follows

$$\boxed{H = -\sum_{i=1}^{N} P_i \cdot \log_2(P_i)}$$

2) Predictive coding

- The idea is to remove the mutual dependency b/w the successive pixels and then perform the encoding.
- Normally the samples would be very large but the difference would be small.

Pixels →

eg - 400 405 420 430

Difference → 5 15 10

300 4096 120 5081 10

frequent variation.

→ This is ~~[illegible]~~ very effectively applied to pulse code and delta modulation category.

3) Transform coding

- objective is to exploit the information packing capability of the transform.
- Energy is packed into fewer components and only these components are encoded and transmitted.
- idea is to remove the redundant high frequency components to create compression.

- Removal of these frequency components leads to loss of information.
- This loss of information, if tolerable can be used for imaging and video applications.

4) Layered coding

- It is useful in case of layered images
- Data structures like Pyramids are useful to represent an image in multiresolution form.
- sometimes these images are segmented and foreground and background and based on the application requirement encoding is done
- It is also in the form of selected frequency coefficients or selected bits ~~of~~ of pixels in an image.

Types of Redundancy (repetitive data)

Redundancy may be implicit or explicit

implicit → cannot remove easily.

explicit ↳ stated very clearly

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

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

not directly expressed but can be observed and can be removed.

eg- AAABB

{(A,3), (B,2)}

$\begin{bmatrix} 10 & 10 & 20 \\ 20 & 20 & 20 \\ 20 & 20 & 10 \end{bmatrix}$

$\boxed{\begin{matrix} 10 & 20 \\ \downarrow & \downarrow \\ 3 & 5 \end{matrix}}$

→ Coding Redundancy
→ Interpixel Redundancy
- Psychovisual Redundancy
- Chromatic Redundancy.

Coding Redundancy

- is caused due to poor selection of coding technique
- assigns a unique code for all symbols of message
- wrong choice of coding creates unnecessary additional bits. These extra bits are called redundancy.

- amount of uncertainty
  self information of event

$$I = \log_2\left(\frac{1}{P}\right)$$

$$= -\log_2(P) \text{ bits}$$

$$\boxed{I \propto \left(\frac{1}{P}\right)}$$

0 — 1

Coding Redundancy = Avg. bits used to code − Entropy

$$\boxed{L_{avg} = \sum_{k=0}^{n} l_{r_k} \cdot P(r_k)}$$

$P(r_k)$ → Probability of pixels

r_k → grey level

l_k → length of code used.

$$H = -\sum_{i=1}^{n} P_i \log_2(P_i)$$

Base of log is 2 → Binary.

Total No. of bits required to code an image of size M X N

= M X N X Lavg.

<u>Inter Pixel Redundancy</u>

- is related with inter pixel correlations within an image
- much of the visual contribution of a single pixel is redundant and can be guessed from the values of its neighbors.

eg – consider an image with a constant background.

- The visual nature of image background is given by many pixels that are not actually necessary.
- This is known as Spatial Redundancy or Geometrical Redundancy

Inter Pixel dependency is solved by algorithms like :

Predictive coding
Bit Plane algorithm
Run length coding
Dictionary based algorithms

Spatial Redundancy may be present in
- single frame (Intra frame)
- or among multiple frames (inter frame or temporal redundancy)

3) <u>Psycho-visual Redundancy</u>

- The eye and the brain do not respond to all information with same sensitivity.

- some information is neglected during the processing by brain. Elimination of this information does not affect the interpretation of image by the brain.

- edges & textural regions are interpreted as important features and the brain groups & correlates such grouping to produce its perception of an object.

Psycho visual redundancy is distinctly vision related and its elimination does result in loss of information

- Quantization is an example
- when 256 levels are reduced by grouping to 16 levels, objects are still recognizable. The ~~pr~~ compression is 2:1 but an objectionable graininess and contouring effect results.

Chromatic Redundancy

- It refers to the presence of unnecessary colors in an image
- The color channels of color images are highly correlated and human visual system cannot perceive millions of colors.
- Therefore the colors that are not perceived by human visual system can be removed without affecting the image quality.

Compression classification

1) Lossless compression - for legal & medical documents, computer programs.
* Exploit only code and interpixel redundancy.

2) Lossy compression - Digital image and video where some errors or loss can be tolerated.
* Exploit both code and interpixel redundancy and Psycho visual perception properties.

Error free compression - is generally composed of two relatively independent operations
1) reduce the interpixel redundancies
2) introduce a coding method to reduce coding redundancies.

- Variable length Coding
- LZW coding
- Bit-Plane Coding
- Lossless Predictive Coding

The most popular technique for removing coding redundancy is due to Huffman (1952)

- It is variable length coding.
- It yields the smallest No. of code symbols per source symbol
- The resulting code is optimal.

Huffman coding

- involves the following 2 steps:-

1) Create a series of source reductions by ordering the probabilities of the symbols and combining the lowest probability symbols into a single symbol and replace in the next source reduction.

2) Each code reduced source starting with the smallest source & working back to the original source.

Algo -

1) Sort the symbols according to decreasing order of their probabilities.

2) Combine the lowest probable source symbols to form a composite symbol with probability equal to the sum of the r

Example

8-bit image 10 x 10 pixels.

Symbol		Probability	1	2	3	4
a_2	40	0.4	0.4	0.4	0.4	0.6
a_6	30	0.3	0.3	0.3	0.3	0.4
a_1	10	0.1	0.1	0.2	0.3	
a_4	10	0.1	0.1	0.1		
a_3	6	0.06	0.1			
a_5	4	0.04				

Huffman source reductions.

original source

Sym.	Prob.	Code	1		2		3		4	
a_2	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
a_6	0.3	00	0.3	00	0.3	00	0.3	00	0.4	1
a_1	0.1	011	0.1	011	0.2	010	0.3	01		
a_4	0.1	0100	0.1	0100	0.1	011				
a_3	0.06	01010	0.1	0101						
a_5	0.04	01011								

Now send $\underbrace{01010}_{a_3}\ \underbrace{011}_{a_1}\ \underbrace{1}_{a_2}\ \underbrace{1}_{a_2}\ \underbrace{00}_{a_6}$

decoded string

$$L_{avg} = \sum_{k=0}^{L-1} l(r_k) \cdot P(r_k)$$

$$= 0.4(1) + 0.3(2) + 0.1(3) + (0.1)(4) + (0.06)5 + (0.04).5$$

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

Entropy $$H = -\sum_{k=0}^{L-1} P(r_k) \cdot \log_2 P(r_k)$$

$$= [0.4 \log_2 0.4 + 0.3 \log_2 0.3 + 0.1 \log_2 0.1 + 0.1 \log_2 0.1 + 0.06 \log_2 0.06 + 0.04 \log_2 0.04)$$

$$= 2.1435 \text{ bits / symbol.}$$

The entropy or avg. information of an image is a measure of the degree of randomness in the image.

- more the entropy means more the randomness or more the uncertainity and thus more information is present.

- It is the lower limit for the average coding length in bits per pixel which can be realized by an ~~optimal~~ optimum coding scheme without any loss of information.

Efficiency of Huffman code

$$\eta = \frac{H}{L_{avg}} = \frac{2.1435}{2.2}$$

$$= 0.9743$$

$$= 97.43\%$$

Compression ratio. $$C_R = \frac{b}{b'} = \frac{4 \times 4 \times 8}{4 \times 4 \times 2.2}$$

$$= 3.63$$

Redundancy $$R = 1 - \frac{1}{C_R}$$

$$= 1 - \frac{1}{3.6}$$

$$= 1 - 0.2754$$

$$= 0.7246$$

$$= 72.46\%$$

Huffman code is lossless & error free and uniquely decodable code.

Arithmetic coding or Range coding

e.g. for medical imaging

- seq. of source symbols are encoded together.
- There is no one-to-one correspondence between source symbols and code words.
- It takes a complete data stream and output one specific code word which is a floating point number between 0 and 1, with as few digits as possible.
- The efficiency = 100%. can always be achieved using arithmetic coding.

as entropy H will be equal to Lavg. It also satisfies Shannon's first theorem which says that minimum coding bits should be equal to entropy H., which is obtained by arithmetic coding.

- It is not practically possible for long messages as arbitrary-precision floating point library / registers are not there.

Steps of Arithmetic Coding

1) Arrange the characters of strings/numbers into ascending order.

2) Divide the numeric range 0 to 1 into the number of different symbols present in the message.

3) Expand the first letter to be coded along with the range

4) Further subdivide this range into No. of symbols.

5) Repeat the procedure ~~th~~ until the termination character is encoded.

- A seq. of source symbol is assigned a single arithmetic code word which corresponds to a sub-interval in [0, 1]

- smaller intervals require more information units (i.e bits) to be represented.

Encode message $a_1\ a_2\ a_3\ a_3\ a_4$

Source symbol	Probability
a_1	0.2
a_2	0.2
a_3	0.4
a_4	0.2

1) Assume message occupies [0, 1)

0 ———————— 1

2) subdivide [0, 1) based on the probability of α_i

3) update interval by processing source symbols.

Initial subinterval

0.2	[0.0, 0.2)
0.2	[0.2, 0.4)
0.4	[0.4, 0.8)
0.2	[0.8, 1.0)

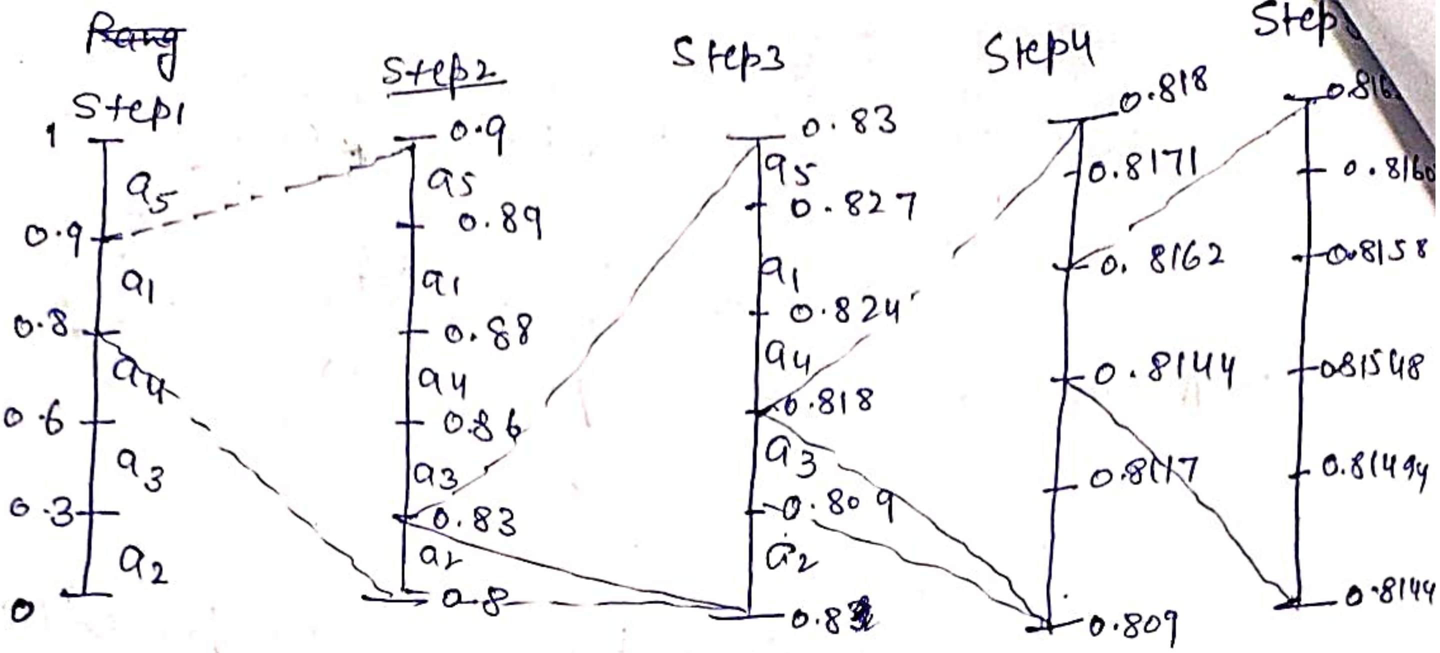
__Example__

a_1	0.1	a_2 0.3	0.0 — 0.3
a_2	0.3	a_3 0.3	0.3 — 0.6
a_3	0.3	a_4 0.2	0.6 — 0.8
a_4	0.2	a_1 0.1	0.8 — 0.9
a_5	0.1	a_5 0.1	0.9 — 1

Range 0 to 1 divided into 5

d = upper bound − lower bound

Range of symbols = lower limit + d (Prob. of symbol)



Step2 (a_1)

$d = 0.9 - 0.8 = 0.1$

range1 $= 0.8 + 0.1(0.3)$
$= 0.8 + 0.03$
$= 0.83$

range2 $= 0.8 + 0.1(0.6)$
$= 0.86$

Step3 (a_2)

$d = 0.83 - 0.80$
$d = 0.03$

range1 $= 0.8 + 0.03(0.3)$
$= 0.8 + 0.009$
$= 0.809$

range2 $= 0.8 + 0.03(0.6)$
$= 0.8 + 0.018$
$= 0.818$

Step4 (a_3)

$d = 0.818 - 0.809$
$= 0.009$

range1 $= \cancel{0.009 + 0.809(0.3)}$
$= 0.809 + 0.009(0.3)$
$= 0.809 + 0.0027$
$= 0.8117$

range2 $= 0.809 + 0.009(0.6)$
$= 0.809 + 0.0054$
$= 0.8144$

Step5 (a_4)

$d = 0.8162 - 0.8144$
$= 0.0018$

Range1 $= 0.8144 + 0.0018(0.3)$
$= 0.8144 + 0.00054$
$= 0.81494$

Range2 $= 0.8144 + 0.0018(0.6)$
$= 0.8144 + 0.00108$
$= 0.81548$

(a5) →

$0.81602 < \text{code} < 0.8162$

Tag = (upper limit of code word + lower limit of code word) / 2

$$= \frac{0.8162 + 0.81602}{2}$$

$$= \underline{0.81611}$$

Decode → $\underline{0.8162}$ (Encoded message)

if $a_1 = 0.1$, $a_2 = 0.3$, $a_3 = 0.3$, $a_4 = 0.2$, $a_5 = 0.1$

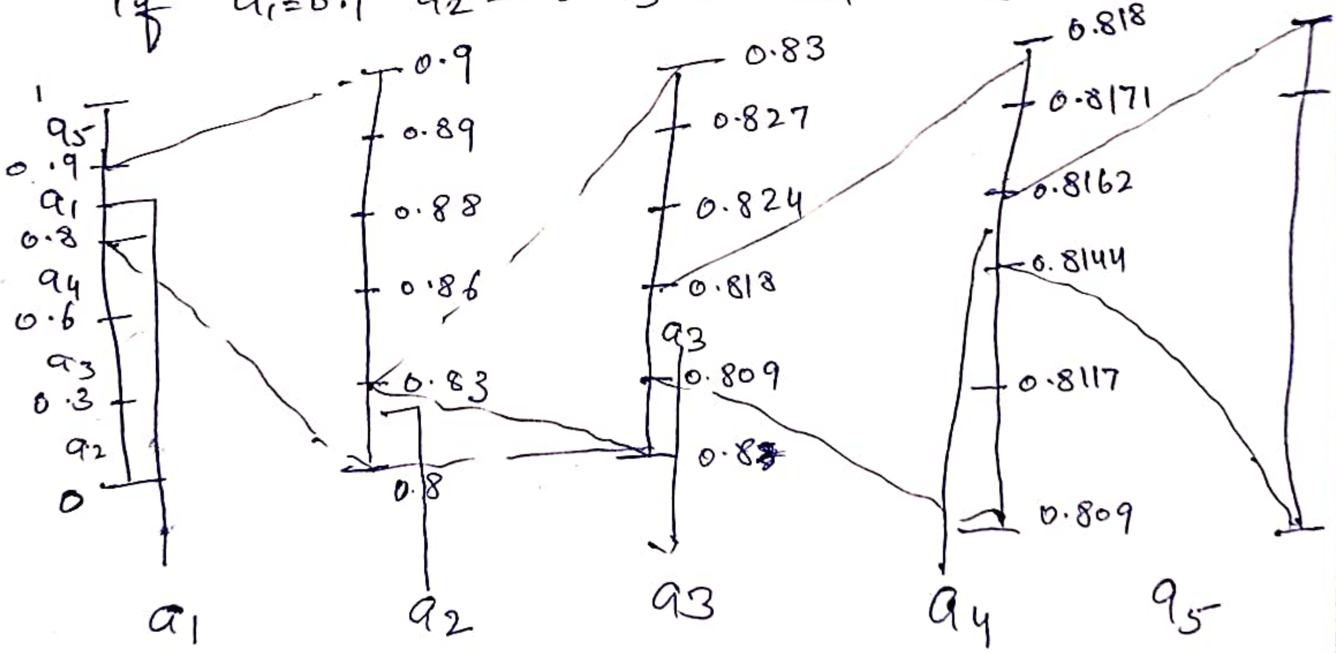


Image segmentation - Input → image
output → attributes

divides an image into regions that are connected and have some similarity within the region and some difference between adjacent regions.

- The goal is usually to find individual objects in an image

- for the most part there are fundamentally two kinds of approaches to segmentation
  - discontinuity
  - similarity

* similarity may be due to pixel intensity, color or texture

* Differences are sudden changes (discontinuities) in any of them but especially sudden changes in intensity along a boundary line which is called an edge.

In compression of data .. represo..

*

Image Segmentation fundamentals

- Segmentation subdivides an image into its constituent regions or objects, until the objects of interest in an application have been isolated.

- segmentation partitions image R into Subregions $R_1, R_2 \ldots R_n$ such that

* $R_1 \cup R_2 \cup \ldots \cup R_n = R$
* Each R_i is a connected set, $i = 1, 2 \ldots n$
* $R_i \cap R_j = \phi$ for all i, j where $i \neq j$
* $Q(R_i) =$ True for every i
* $Q(R_i \cup R_j) =$ False for ~~two~~ any 2 adjacent regions.

- segmentation Problem - To partition the image into regions satisfying above conditions.

Two principal approaches

- Edge-based segmentation
  * Partition an image based on abrupt change in intensity (edges)

- Region-based segmentation
  * partition an image into regions that are similar according to a set of predefined criteria.

Image segmentation Techniques

- Discontinuity (Edge based)
  - edge based (Point, Line & Edge)
    - First order: Gradient based
      - Sobel
      - Robert
      - Prewitt
    - Second order: Gaussian based
      - Canny edge detection
      - Laplacian of Gaussian
- similarity (Region based)
  - Clustering based
  - Thresholding
    - Global
    - Local
  - Region-based
    - Region Growing
    - Region Splitting & Merging

Detection of 3 basic types of grey-level discontinuities

- Points, lines, edges

- use the image sharpening techniques
  - * The 1^{st} order derivatives produce thicker edges.
  - * 2^{nd} order derivatives (Laplacian operator)

a strong response to fine details, such as thin lines and isolated points & noise.

- can be done by running a mask through the image.

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9$$

$$= \sum_{k=1}^{9} w_k z_k$$

z_1	z_2	z_3
z_4	$\boxed{z_5}$	z_6
z_7	z_8	z_9

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

This is known as convolution.

Point detection

steps

-1	-1	-1
-1	8	-1
-1	-1	-1

1) Apply Laplacian filter to the image to obtain $R(x,y)$.

2) Create binary image by threshold

$$g(x,y) = \begin{cases} 1 & \text{if } |R(x,y)| \geq T \\ 0 & \text{otherwise} \end{cases}$$

where T is a non negative threshold.

line detection

- A special mask is needed to detect a special type of line

eg.- Horizontal mask has high response when a line is passed through the middle row of the mask.

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal.

-1	-1	2
-1	2	-1
2	-1	-1

$45°$

-1	2	-1
-1	2	-1
-1	2	-1

Vertical

2	-1	-1
-1	2	-1
-1	-1	2

$-45°$

mpression / ...ble. ...ing ... as ...ntity ...ron
...ent

edge detection - is the approach for segmentation images based on abrupt changes in intensity. It used to detect the boundaries or to find size or location of an object in an image.

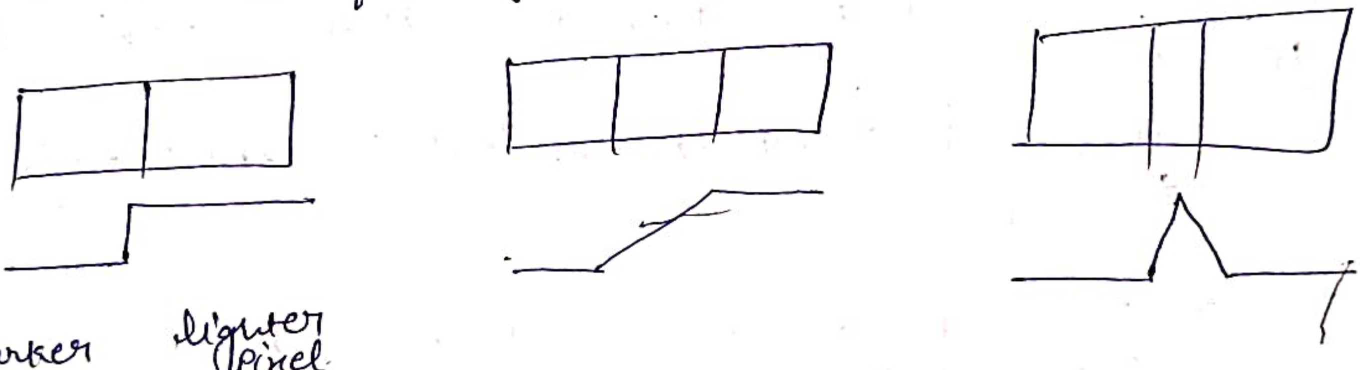
what is an edge

- an edge is a set of connected pixels that lie on the boundary between two regions.

- an edge is a 'local' concept whereas a region boundary, owing to the way it is defined, is a more global idea.

Edge models.

1) - step edge (ideal edge)
2) - ramp edge (thick edge)
3) ~~and~~ roof edge.



Darker Pixel   lighter pixel

In edge detection we convolve a filter (Mask) with the image.

$$\begin{bmatrix} 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \\ 10 & 10 & 10 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 30 & 30 & 0 \\ 0 & 30 & 30 & 0 \\ 0 & 30 & 30 & 0 \\ 0 & 30 & 30 & - \end{bmatrix}$$

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

- Horizonal edge kernel.
- Vertical edge kernel — $\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$
- outline kernel $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$
- Emboss kernel

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

Edge detection operators — ~~bas~~ Guassian Based

Gradient Based operator which computes first order derivations in a digital image like

* Sobel
* Prewitt
* Robert

Based on convolution only masks are different

Sobel

vertical direction

$$\begin{bmatrix} 1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$
G_x

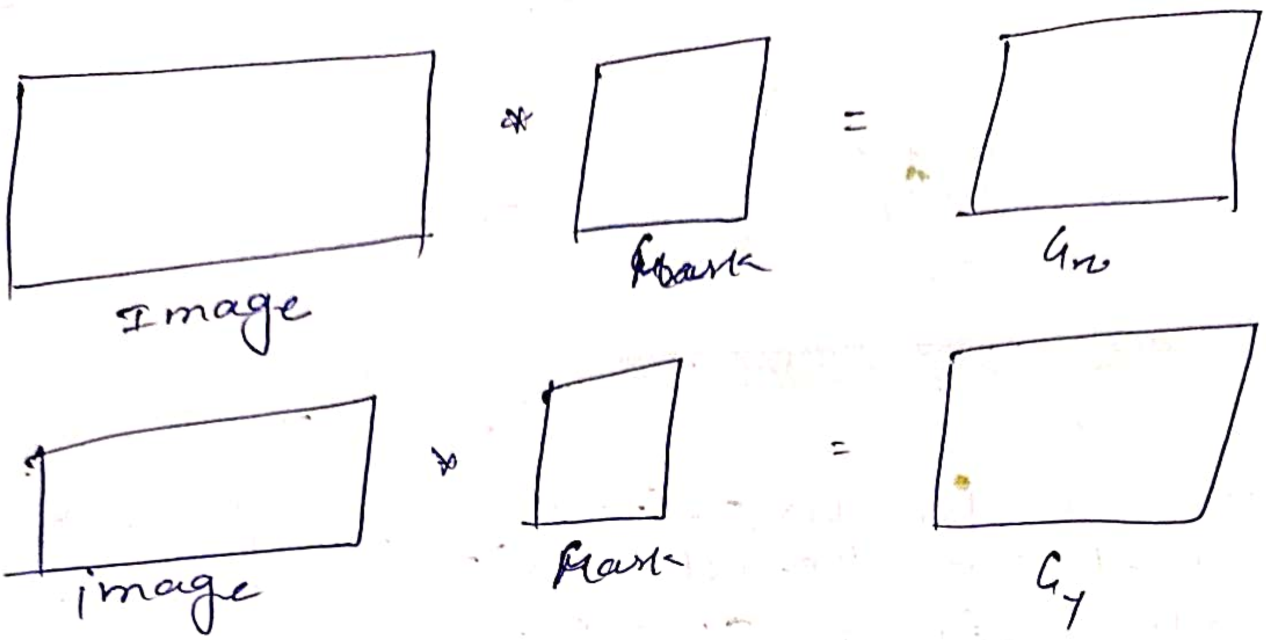
we can apply more weight to masks

$$\begin{bmatrix} -1 & 0 & 1 \\ -5 & 0 & 5 \\ -1 & 0 & 1 \end{bmatrix}$$

Horizontal

$$\begin{bmatrix} 1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$
G_y

$$\boxed{|G| = \sqrt{G_x^2 + G_y^2}}$$



2) Prewitt operator → This is combination of horiz edge detector & vertical edge dector.

Vertical direction

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

Horizontal Direction

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

3) Roberts cross edge detector

2×2 convolution

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

$$G = \sqrt{G_x^2 + G_y^2}$$