28/3/16

MODULE - 6

IMAGE COMPRESSION.

Data compression refers to the process of reducing the amount of data required to represent a given quantity of information. Data are the means by which the information is conveyed. Because various amounts of data can be used to represent the same amount of information, representations that contain irrelevant or repeated information are said to contain redundant data. If b and b' denote the no. of bits in two representations of the same information, relative data redundancy R is defined as

$$R = 1 - \frac{1}{c}$$, where c is the compression ratio. It is defined as

$$c = \frac{b}{b'}$$

The 2D intensity arrays (images) suffer from 3 principal types of data redundancies defined as

1) Coding redundancy

A code is a system of symbols (numbers, bits, etc..) used to represent a body of information.

Each piece of information is assigned a sequence of code symbols called code words. The no. of symbols in each code word is its length. The 8 bit codes that are used to represent the intensities may contain more bits than needed.

Let ~~the~~ r_k denotes a discrete random variable in the interval $(0, L-1)$ is used to represent the ~~the~~ intensities of an $M \times N$ image and the probability of occurance of r_k is defined as

$$P_r(r_k) = \frac{n_k}{MN}$$

where n_k is the no. of times r_k occurs

~~when~~ The average no. of bits used to represent each pixel is $L_{avg} = \sum_{k=0}^{L-1} l(r_k) P_r(r_k)$

where $l(r_k)$ is the no. of bits used to represent each value of r_k.

Eg:

Probability of occurance	Code1	Code 2
0.25	01101011	01
0.41	00000011	1
0.34	01101111	110

Code1

$L_{avg} = 8 \times 0.25 + 8 \times 0.41 + 8 \times 0.34 - \underline{\underline{8}}$

Code2

$L_{avg} = 2 \times 0.25 + 1 \times 0.41 + 3 \times 0.34 = \underline{\underline{1.9}}$

let $b = 8$ and $b' = 1.9\cancel{7}$

$$c = \frac{b}{b'} = \underline{\underline{4.21}}$$

$$R = 1 - \frac{1}{c} = \underline{\underline{0.7}}$$

2) Irrelevant information

Most of the images contain information that is ignored by the human visual system. It is redundant in the sense that it is not used because this results in a loss of quantitative information, it's removal is commonly referred to as quantization.

3) Spatial or Temporal Redundancy.

Because the pixels of most 2D intensity arrays are correlated spatially (each pixel is similar to or dependent on neighbouring pixels) the information is unnecessarily replicated in the representations of the correlated pixels. This is what is known as spatial or temporal redundancy. To reduce the redundancy associated with spatially and temporily correlated pixels, an image must be transformed into a more efficient form called mappings. A mapping is said to be reversible if the pixels of the original image can be reconstructed without error from the transformed data set.

Measuring Information

In information theory, a random event E with probability P(E) is said to contain

$$I(E) = \log \frac{1}{P(E)} = -\log P(E) \qquad [\log 1 = 0]$$

amount of information. If $P(E) = 1$, then $I(E) = 0$. which means no information is conveyed. Given a source of independent random events which produces a set of events $a_1, a_2, a_3, \ldots$ with associated probabilities $p(a_1), p(a_2) \ldots$ the average information per source o/p called entropy is given by

$$H = -\sum_{j=1}^{J} P(a_j) \log (P(a_j))$$

Shannon's 1st theorm

$$\lim_{n \to \infty} \frac{L_{avg}}{n} = H$$

Fidelity criteria.

When information loss can be expressed as a mathematical function of the i/p and o/p of a compression process, it is said to be based on an objective fidelity criterion.

Let $f(x,y)$ be an i/p image and $\hat{f}(x,y)$ be an approximation of $f(x,y)$ that results from compressing and decompressing the i/p.

The error $e(x,y)$ is given by,

$$e(x,y) = \hat{f}(x,y) - f(x,y)$$

The total error between two images is given by $e(x,y) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y) - f(x,y)\right]$

The root mean square error

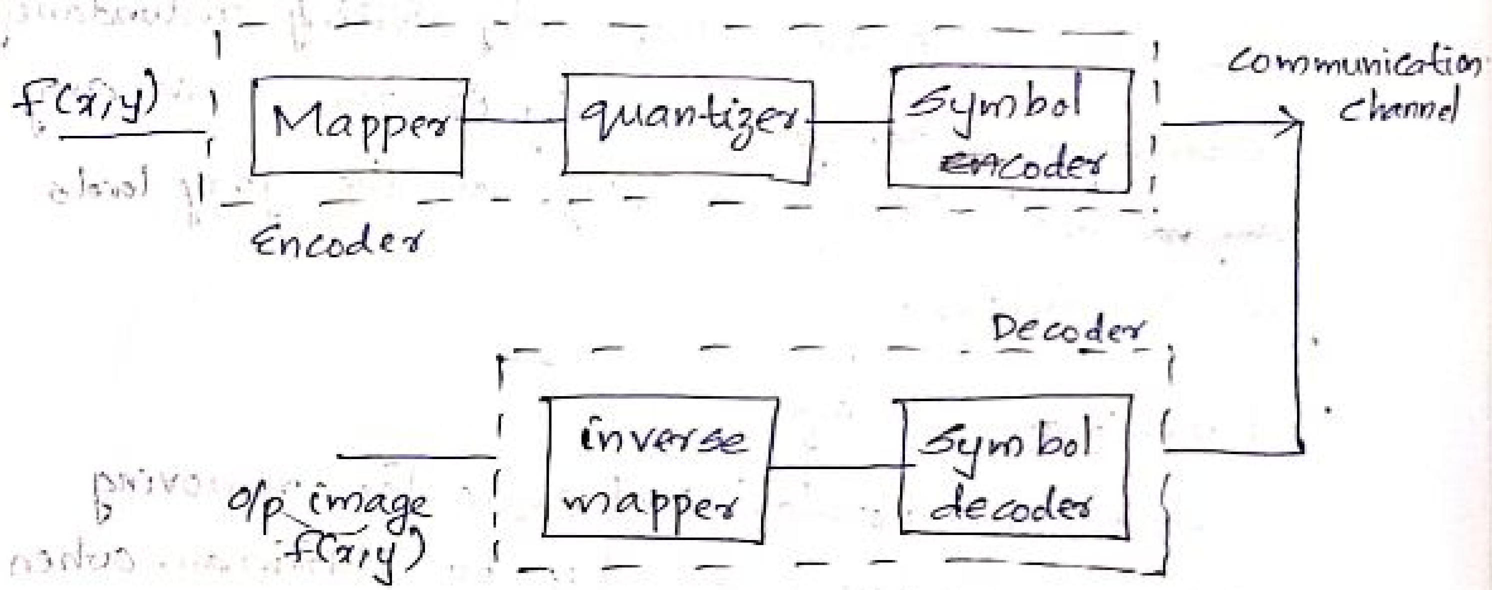
$$e_{rms} = \left[\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y) - f(x,y)\right]^2\right]^{1/2}$$

Signal to noise ratio, $SNR = \dfrac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y)\right]^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[\hat{f}(x,y) - f(x,y)\right]^2}$

Scanned with CamScanner

Image Compression Model

An image compression system consists of two distinct functional components, an encoder and a decoder. An encoder performs compression and a decoder performs decompression.



Encoder consists of a mapper, quantizer and a symbol coder.

* A mapper transforms an image into a format designed to reduce spatial and temporal redundancy. This operation is reversible. Run length coding is an example of a mapping technique.
* The quantizer reduces the amount of irrelevant informations out of the compressed representation. It is an irreversible process.
* symbol coder generates a fixed or variable length

code to represent the quantizer output. In most cases it uses a variable length code which reduces coding redundancy.

Error free Compression (variable length coding)

The simplest approach to error-free image compression is to reduce only coding redundancy. Coding redundancy normally is present in any ~~any~~ natural binary encoding of the gray levels in an image.

. Huffman Coding.

The most popular technique for removing coding redundancy is due to Huffman. when coding the symbol of an information source individually, Huffman coding yields the smallest possible number of code symbols per source symbol. The resulting code is optimal for a fixed value of n, subject to the constraint that the source symbols be coded one at a time.

* The first step in Huffman's approach is to create a series of source reductions by ordering the probabilities of the symbols under consideration and combining the lowest probability symbols

into a single symbol that replaces them in the next source reduction.

* The second step in Huffman's procedure is to code each reduced source, starting with the smallest source and working back to the original source. The minimal length binary code for a two-symbol source is the symbols 0 and 1.

Eg:- Step 1 :-

Original source		Source reduction			
Symbol	probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	0.4
a_1	0.1	0.1	0.2	0.3	
a_4	0.1	0.1	0.1		
a_3	0.06	0.1			
a_5	0.04				

Step 2

Original source			Source reduction							
Symbol	probability	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								

The average length of the code is

$$L_{avg} = 0.4 \times 1 + 0.3 \times 2 + 0.1 \times 3 + 0.1 \times 4 + 0.06 \times 5 + 0.04 \times 5$$
$$= 2.2 \text{ bits/symbol}.$$

Huffman's procedure creates the optimal code for a set of symbols and probabilities subject to the constraint that the symbols be coded one at a time. After the code has been created, coding and/or decoding is accomplished in a simple lookup table manner. The code itself is an instantaneous uniquely decodable block code. It is called a block code because each source symbol, is mapped into a fixed sequence of code symbols. It is instantaneous, because each code word in a string of code symbols can be decoded without referencing succeeding symbols. It is uniquely decodable, because any string of code symbols can be decoded in only one way.

Compression schemes

1) lossless → same amount of data is retrieved without distortion.

2) lossy → receiver side some distortion occurs

Need for compression

4/4/18. 1) Dictionary based Compression.

Dictionary based compression uses a family of algorithms that encode variable length patterns of symbols as an index to the same pattern in a dictionary or lookup table. If this index is smaller than the actual length of the pattern, then compression occurs. The dictionary used for compression can be static or dynamic. Most dictionary based schemes are based on Lempel & Ziv algorithm, which was developed by Jacob Ziv and Abraham Lempel. The algorithms are known as LZ77 and LZ78 which are the basis of all compression used in lossless applications. It is used in Gif type file formats.

LZ77 Algorithm

Here the data to be encoded is passed over by a window. The window is divided into two sections. The 1st section consists of a large block of recently encoded data. 2nd section is a smaller section termed as look ahead buffer. The algorithm tries to match the contents of the look ahead buffer to a string in the dictionary.

LZW8 Algorithm:

It takes a different approach to coding a digital signal to get around the problems in LZW algorithm.

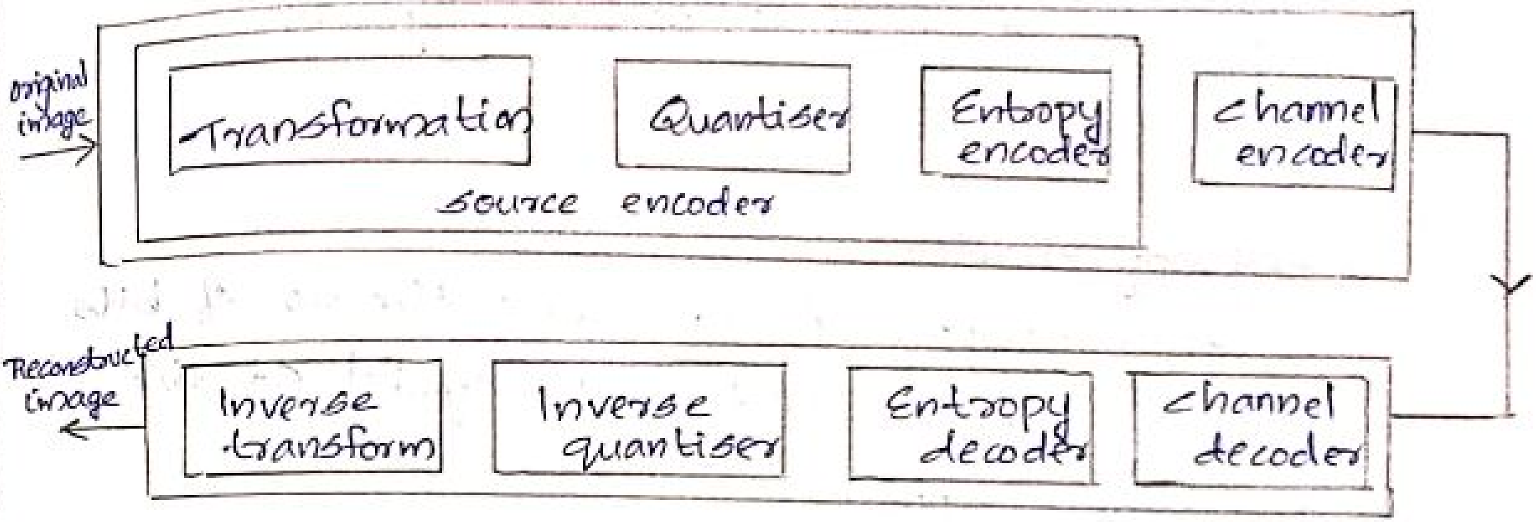
2) Transform based compression

In image transform coding, an image is linearly transformed to produce a set of linear transform coefficient which are usually scalar quantised and entropy coded for transmission. This transform coding is a mathematical operation that converts a large set of highly populated pixels into a smaller set of uncorrelated coefficients.

Transform coding relies on the fact that pixels in an image exhibit a certain level of correlation with their neighbouring pixel. Transformation is used to transform the image data in time domain to frequency domain, thus spacial redundancy can be minimized. Thus the energy of the transformed data is mainly condensed in the low frequency region and is represented by a few transform coefficients.

Scanned with CamScanner

Thus most of these coefficients can be discarded without significantly affecting the reconstructed image quality.



Transformation

For efficient compression, the transform should have the following properties.

1) Decorrelation

The transform should generate less correlated or uncorrelated transform coefficients to acheive high compression ratio.

2) Linearity

This principle allow one to one mapping between pixel values and transform coefficients.

3) Orthogonality

orthogonal transform have the feature of eliminating redundancy in the transformed image.

Quantisation

Reduces the number of bits needed to represent a data. It is basically an irreversible process.

Entropy Encoder

The purpose is to reduce the no. of bits required to represent each symbol at the quantizer output.

Commonly used entropy techniques are Huffman coding, Arithmetic coding, Run length coding. It is basically a lossless compression scheme.

The output of the encoder is a bit stream which is transmitted through the channel to the decoder. The channel is usually assumed to be lossless. The decoder basically performs the reverse process of encoder to get the reconstructed image.

3). Wavelet based Image Compression

The wavelet transform decomposes an image into a set of difference resolution sub images corresponding to various frequency bands. This results in multiresolution representation of images with localisation in both spatial and frequency domain. The main advantage is

(i) wavelets have non uniform frequency spectra, which facilitate multiscale analysis.

(ii). The multiresolution property can be used to exploit the fact that the response of the human eye is different to high and low frequency components of an image.

(iii) Discrete wavelet transform (DWT) can be applied to entire image. which reduces blocking artifacts.

DWT can be implemented through

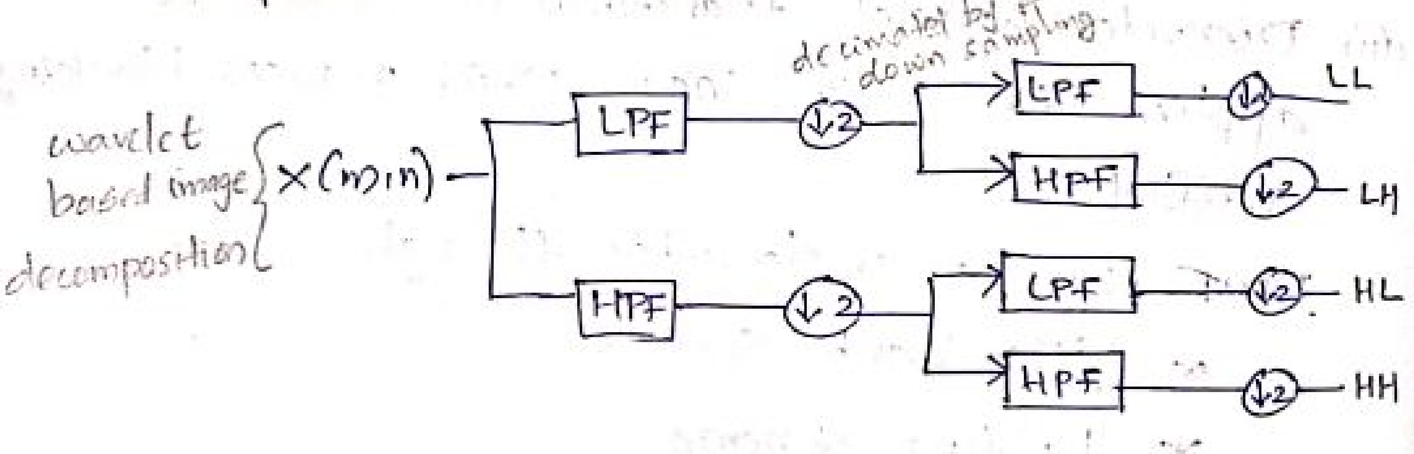
* Filter bank scheme
* Lifting scheme

① Filter bank scheme

* Sub bank coding is a procedure in which i/p signal is subdivided into several freq. bands.

* Sub bank can be implemented through a filter bank.
* Filter bank is a collection of filter having either a common i/p or common o/p.
* with common i/p they form an analysis bank and with common o/p they form synthesis bank.
* A complete two channel filter bank is composed to two filter bank ie, analysis and synthesis section.

The analysis section decomposes the signal into a set of sub band components and the synthesis section reconstructs the signal from its components.



The describable characteristics of a filter bank includes :-

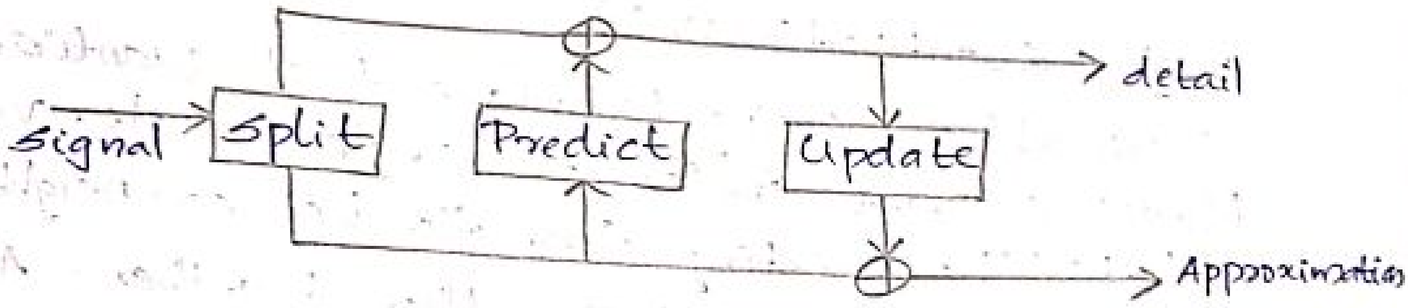
1) Maximal decimation
2) Seperable filtering
3) Polyphase form

4) Perfect reconstruction
5) Pre structure decomposition

② Lifting Scheme

It was developed as a method to improve wavelet transform. Lifting procedure consists of 3 phase

(i) split phase
(ii) Predict phase
(iii) Update phase



The 1st step in lifting scheme is to separate the original sequence x into two subsequences containing the odd indexed samples x_o and even indexed samples x_e. This subsampling step is also called lazy wavelet transform which is given by,

$$x_o : d_i \leftarrow x_{2i+1}$$

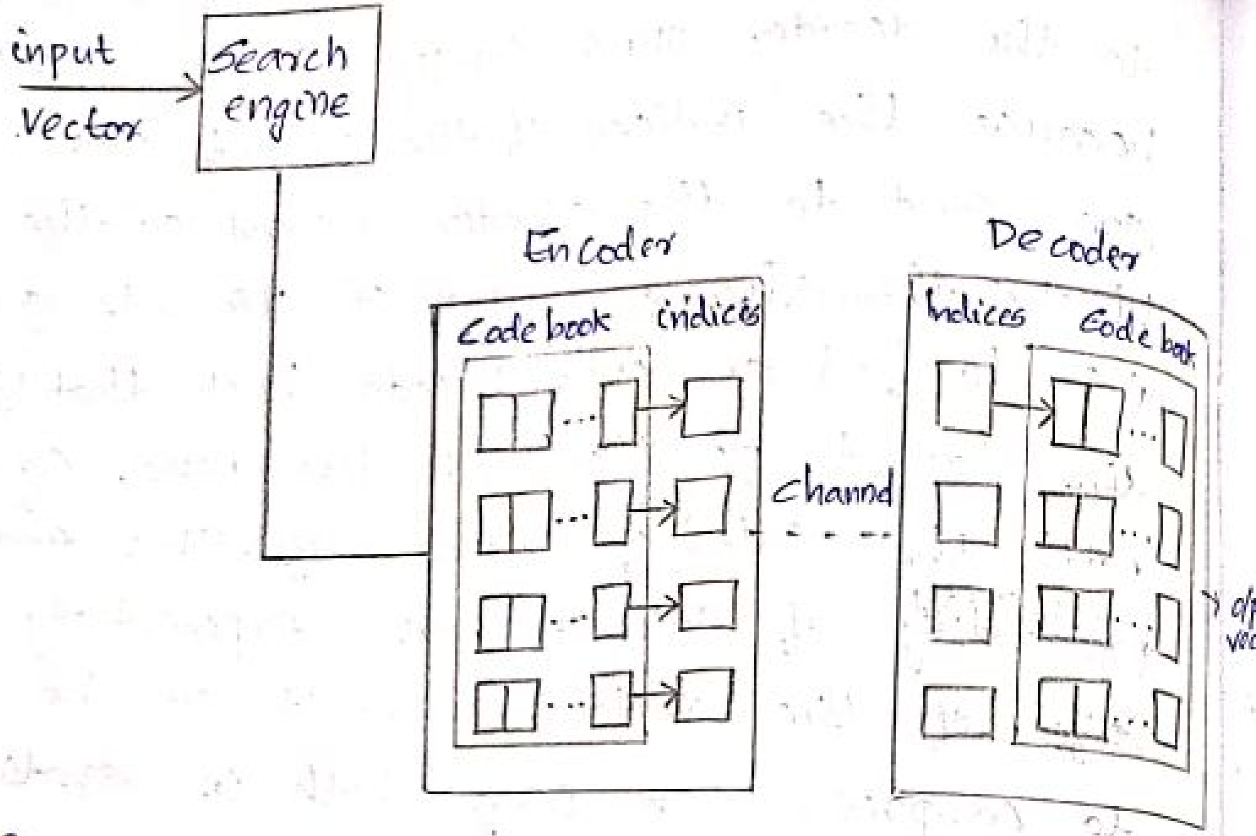
$$x_e : s_i \leftarrow x_{2i}$$

x_e and x_o are highly correlated. Hence a predicator P can be used to predict one set from another. This is called dual lifting. Here the odd samples are predicted using the neighbouring even indexed sample and prediction error is recorded replacing the original sample values. The even samples are replaced with smoothed values using the update operator U. The o/p from the dual lifting step (P) provides lowpass filtered version of i/p and after U we get high passed filtered version of i/p

Vector Quantisation (VQ)

It is a block coding technique that quantises blocks of data instead of single sample. VQ exploits the existing correlation between neighbouring slg samples by quantising them together. A VQ scheme can be divided into two parts: Encoding procedure and decoding procedure. The i/p image is partitioned into a set of non overlapping image blocks. The closest codeword in the code book is then found for each image block. The closest codeword is the one in the code book that has a minimum squared Euclidean from the i/p block. The corresponding index for each searched closest word is transmitted to the decoder. Thus compression is acheived because the indices of the closest code words are send to the decoder instead of the image blocks themselves. The goal of VQ code generation is to find an optimal code book that yields the lowest possible distortion when compared with the code book of same size. The ~~conditional~~ computational ~~probability~~ complexity of VQ increases exponentially with size of the vector blocks. It can be used to compress an image both in spatial and freq. domain. VQ is ~~the~~ a lossy compression scheme based on principles of block coding.

* A vector quantizer maps a dataset in a N dimensional space into a set of finite vectors. Each vector is called a code vector or a code word. The set of all code words is called a code book. Each i/p vector can be associated with an index of a code word and this index is transmitted instead of the original vector. This index can be decoded to get the code word that is represented.



11/4/18

Huffman coding problem

Q. Obtain the Huffman code for the word COMMITTEE

Ans:-

* Arrange the probability in descending order

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
C – 1/9	M – 2/9 01	2/9 01	3/9 00	4/9 1	5/9 0
O – 1/9	T – 2/9 10	2/9 10	2/9 01	3/9 00	4/9 1
M – 2/9	E – 2/9 11	2/9 11	2/9 10	2/9 01	
I – 1/9	C – 1/9 001	2/9 000	2/9 11		
T – 2/9	O – 1/9 0000	1/9 001			
E – 2/9	I – 1/9 0001				

Final code word is

M - 01
T - 10
E - 11
C - 001
O - 0000
I - 0001

o/p vector

Average bit length, $L = \sum_{k=0}^{N-1} P_k l_k$

$L = 2\times\frac{2}{9} + 2\times\frac{2}{9} + 2\times\frac{2}{9} + 3\times\frac{1}{9} + 4\times\frac{1}{9} + 4\times\frac{1}{9}$

$= \frac{4}{9} + \frac{4}{9} + \frac{4}{9} + \frac{3}{9} + \frac{4}{9} + \frac{4}{9} = \frac{23}{9} = 2.555$

$= 2.555$ bits/symbol

Entropy, $H = -\sum_{k=0}^{N-1} P_k \log_2 P_k$

$\log_2 P_k = \frac{\log_{10} P_k}{\log 2}$

$\therefore H = \frac{1}{\log 2} \sum_{k=0}^{N-1} P_k \log_{10} P_k =$

$= -3.32\left[\frac{2}{9}\log\left(\frac{2}{9}\right) + \frac{2}{9}\log\left(\frac{2}{9}\right) + \frac{2}{9}\log\left(\frac{2}{9}\right) + \frac{1}{9}\log\left(\frac{1}{9}\right) + \frac{1}{9}\log\left(\frac{1}{9}\right) + \frac{1}{9}\log\left(\frac{1}{9}\right)\right]$

$= -3.32[-0.435 - 0.318] = 2.499$ bits/symbol.

Scanned with CamScanner

$$\text{efficiency}, \ \eta = \frac{H}{L}$$

$$= \frac{2.499}{2.55} \times 100 = \underline{\underline{98\%}}$$

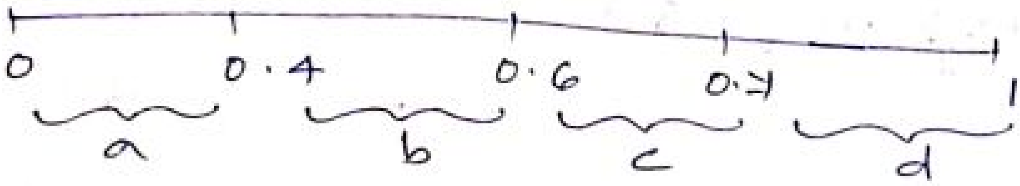
Arithmetic Coding.

It is non binary.

In arithmetic coding, the interval from 0 to 1 is divided according to the probabilities of the occurance of probabilities of intensities. It performs arithmetic operations on a block of data based on the probabilities of the next character.

Q) A Source emits 4 symbols {a, b, c, d} with Probabilities 0.4, 0.2, 0.1 and 0.3 respectively. Construct arithmetic coding to encode the word dad.

Ans:- Step1:- Our objective is to encode the word dad. Initially the range of intervals is assumed to be between 0 and 1. The individual symbol range is as shown below.



Step2:- The first symbol to be transmitted is d.
Hence the new range is from 0.7 to 1, so we need to find the subrange for each symbol.

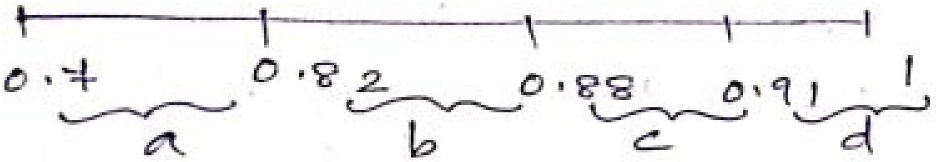
The low range of symbol a is 0.7

High range of a = $0.7 + 0.3 \times 0.4 = 0.82$
= low range of symbol b

High range of b = $0.7 + 0.3 \times 0.6 = 0.88$
= low range of symbol c

High range of c = $0.7 + 0.3 \times 0.7 = 0.91$
= low range of symbol d

high range of d = 1



Step3:- The next symbol to be transmitted is a.
The range of a is from 0.7 to 0.82.
To find the sub intervals,
low range of a is 0.7
high range of a = $0.7 + 0.12 \times 0.4$
= 0.748 = low range of b

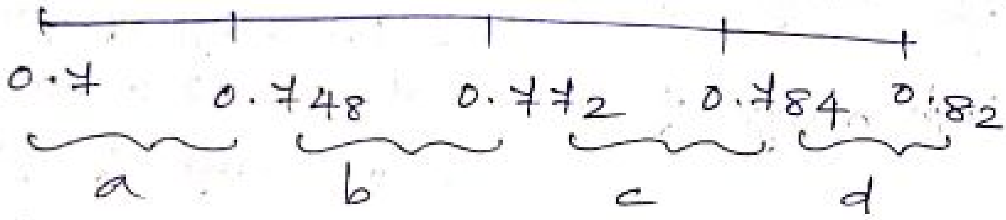
High range of b = $0.7 + 0.12 \times 0.6 = 0.772$

= Low range of c

High range of c = $0.7 + 0.12 \times 0.7 = 0.784$

= Low range of d

High range of d = 1



The interval of next symbol to be transmitted is 0.784 to 0.82

The tag corresponding to the transmitted is

$$\frac{0.784 + 0.82}{2} = \underline{0.802}$$

Image Compression Standards.

refer text

JPEG

MPEG