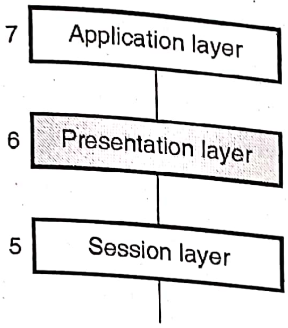
1 Presentation Layer :

- The presentation layer is the 6th layer the OSI model as shown in Fig. 9.1.1.



(G-707) Fig. 9.1.1 : Position of presentation layer

- Above it there is the application layer and below it there is the sessions layer.
- The presentation layer is related to the **syntax** and **semantics** of the information being exchanged between the interested systems.
- Some of the important responsibilities of the presentation layer are :

  1. Translation 2. Encryption 3. Compression.

1. Translation :

- The communication systems usually exchange the information in the form of strings of characters, numbers etc.
- This information needs to be changed into bit streams before transmission.
- This is essential because different systems use different encoding techniques. The presentation layer does the job of translation.
- The presentation layer at the sending end converts the information into a common format and the presentation layer at the receiving end will convert this common format into the one which is compatible to the receiver.

2. Encryption :

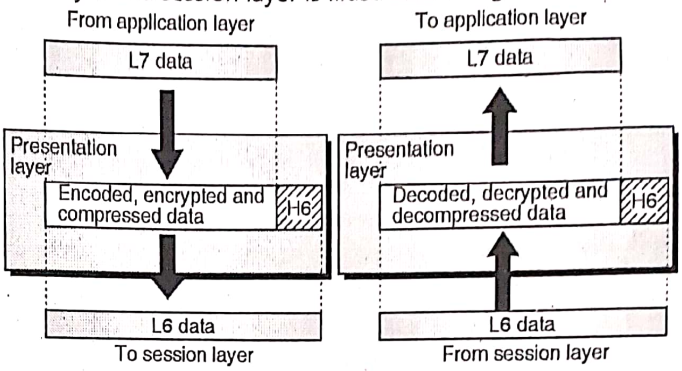
- For ensuring the security and privacy of the information that is being communicated, a process called data encryption is essential.
- Encryption is carried out at the sending end. In the encryption process, the sender transforms the original information to another form, and sends the transformed information.
- At the receiving end, an exactly opposite process called Decryption, is carried out in which the received information is transformed back to its original form.
- Both encryption and decryption are carried out by the presentation layer.

3. Compression :

- The data compression technique is used for reducing the number of bits required to send an information.
- Data compression is essential for transmission of multimedia such as text, audio and video.

Relation with application and session layers :

- The relation of presentation layer with the application layer and session layer is illustrated in Fig. 9.1.2.



(G-69) Fig. 9.1.2 : Relation of presentation layer with the application layer and session layer

- The data from the application layer (L7 data) is encrypted, encoded and compressed at the presentation layer.
- A presentation layer header H-6 is also added as shown in Fig. 9.1.2.
- This is then sent to the session layer as L-6 data. These processes take place at the sending end of the system.
- While receiving the data from session layer, the operations carried out by the presentation layer are exactly opposite to those carried out while transmitting.
- The received data from the session layer undergoes decryption, decompression and decoding at the presentation layer.
- The header H-6 is detached from the data and then the L-7 data is sent to the application layer.

Functions of presentation layer :

- The presentation layer performs the following function :
  1. It translates data between the formats the network requires and the format the computer expects (e.g. ASCII or EBCDIC).
  2. It does the protocol conversion.
  3. For security and privacy purpose, it carries out encryption at the transmitter and decryption at the receiver.
  4. It carries out data compression to reduce the bandwidth of the data, to be transmitted.
- Unlike the session layer, which provides many different functions, the presentation layer has only one function.

- It primarily functions as a pass through device. It receives primitives from the application layer and issues duplicate primitives to the session layer below it, using the Presentation Service Access Point (PSAP) and Session Service Access Point (SSAP).

9.2 Need of Data Compression :

MU : May 14, May 15, May 16

University Questions

Q. 1 What is compression ? List compression algorithms. **(May 14, May 15, May 16, 4 Marks)**

Q. 2 What is compression ? List different compression algorithm. Why adaptive Huffman coding is used ? **(May 15, 4 Marks)**

Definition :

- Data compression is defined as a compact representation of data.
- Now a days a huge amount of data is exchanged over the Internet, mobile phones and other data network.
- Due to the increased demand of digital video content, the amount of data exchange has increased further.
- The **bandwidth** requirement increases with increase in the amount of data.
- However it is not possible to increase bandwidth infinitely.

Need :

- In order to transmit and receive the audio, video and data without increasing the bandwidth much and without degrading the quality of video and audio contents, we have to use some **data compression** technique.
- Application areas of data compression are : Audio/Video content streaming on the Internet, Long distance mobile communication, Listening music on MP-3 player or watching movies on DVDs.

Reasons :

- The reason why data compression is needed are :

1. **Downloading uncompressed data needs time :**

- Now-a-days all the information such as voice, images, videos and data is in the digital form.
- We need a huge data to represent the video information. (e.g. a one second uncompressed video needs 20 Mega bytes or 160 Megabit data).
- We will need very large discs to contain the data corresponding to one full HD quality movie.
- Even the time required, to download the music or video is too long.

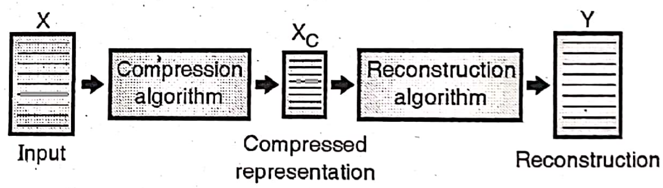
2. Storage of uncompressed data needs huge space.
3. The bandwidth required for the transmission of uncompressed data is very large.

9.3 Compression and Reconstruction :

- The compression techniques are also called as **compression algorithms.**
- There are two types of such algorithms :
  1. Compression algorithms.
  2. Reconstruction algorithms.
- A **compression algorithm** takes an input X and produces its compressed representation X_C which needs less number of bits than X itself.
- On the other hand, a **reconstruction algorithm** takes in the compressed representation X_C and works on it to produce the reconstruction Y.
- As per convention, the compression and reconstruction algorithms together are referred to as the **compression algorithm.**

Compression Algorithm = Compression Algorithm + Reconstruction Algorithm

- The concept of compression and reconstruction, has been demonstrated in Fig. 9.3.1.



(O-423) Fig. 9.3.1 : Compression and reconstruction

9.4 Classification of Compression Techniques :

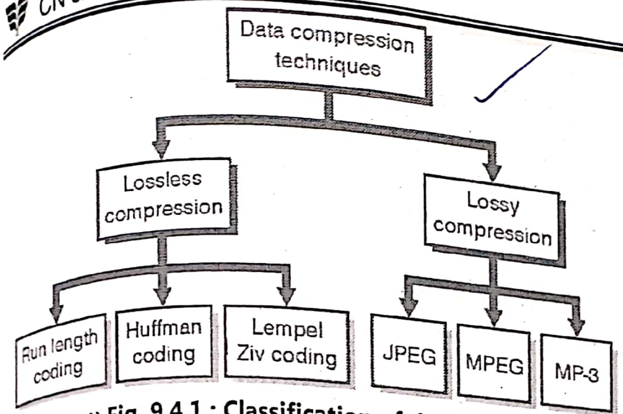
MU : May 14, May 15, May 16

University Questions

Q. 1 What is compression ? List compression algorithms. **(May 14, May 15, May 16, 4 Marks)**

Q. 2 What is compression ? List different compression algorithm. Why adaptive Huffman coding is used ? **(May 15, 4 Marks)**

- Depending on the need of reconstruction, we can divide the data compression techniques into two types :
  1. Lossless compression techniques
  2. Lossy compression techniques.
- The classification of data compression techniques is shown in Fig. 9.4.1.



(0-424(a)) **Fig. 9.4.1 : Classification of data compression techniques**

- The lossless compression techniques are further classified into three types : Run Length Encoding (RLE), Huffman coding and Lempel-Ziv coding.
- The lossy compression techniques, are further classified into three types : JPEG (For images), MPEG (For video) and MP-3 (For sound).

9.4.1 Lossless Compression : **MU : Dec. 15**

University Questions

Q. 1 What is lossless compression ? **(Dec. 15, 5 Marks)**

Definition :

- The lossless compression is the type of data compression technique in which only the redundant information contained in the data is removed.
- Due to removal of such information, there is no loss of the data, which contains information. Hence it is called as **lossless compression.**
- Lossless compression is also known as **data compaction.**
- Lossless compression is achieved using those techniques, which generate an exact duplicate of the input data stream after following compression at transmitter and expansion at the receiver.
- Therefore, the data integrity is preserved in lossless compression.
- The compression and decompression algorithms are exactly opposite to each other. In this process there is no loss of the data which contains information.
- Lossless compression is used, when we cannot afford any data loss. In some lossy compression processes for further reduction of data size, lossless compression is applied , as the last step.

9.4.2 Lossy Compression : **MU : Dec. 15**

University Questions

Q. 1 Describe Lossy compression methods. Where we use Lossy compression methods ? **(Dec. 15, 6 Marks)**

Definition :

- This is the class of data compression methods, that use inexact approximations and partial data discarding to represent the content.
- Thus in this type of compression, there is a loss of information in a controlled manner, without a significant loss of important information.
- The lossy compression is therefore not completely **reversible.**
- That means we cannot reproduce the exact replica of the original picture at the receiver.
- But the advantage of this type is **higher compression ratios** than the lossless compression.
- The lossless compression is used for the digital data.
- The lampel - Ziv algorithm is used for the same.

Applications :

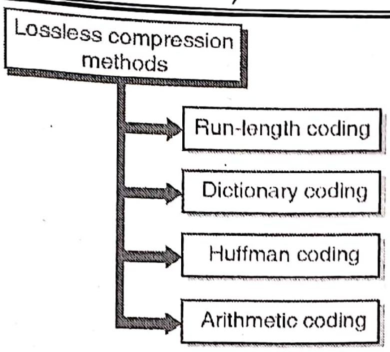
- For many applications, the lossy compression is preferred due to its higher compression without a significant loss of important information.
- Lossy compression can be applied to graphical images and digitized voice.
- By their very nature, these digitized representations of analog phenomenon are not perfect and so a small loss of information due to compression does not matter much.
- Lossy compression is preferred for compressing the audio and video contents, where a slight loss of information does not affect the quality much.
- There is limit on the amount of compression in lossless compression methods.
- In the following subsections we will discuss some lossy compression methods.

9.4.3 Types of Lossless Compression :

- Data integrity is preserved in lossless compression.
- The compression and decompression algorithms are exactly opposite to each other.
- In this process there is no loss of the data which contains information.
- Lossless compression is used when we cannot afford any data loss.
- In some lossy compression processes for further reduction of data size, lossless compression is applied as the last step.

Lossless compression methods :

- Lossless compression methods are as shown in Fig. 9.4.2.



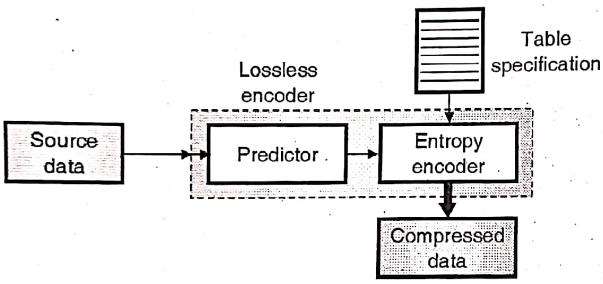
(G-2263)

Fig. 9.4.2 : Lossless compression methods

- Lossless compression techniques are generally preferred to compress the **text data.**

Block diagram :

- Fig. 9.4.3 shows a simplified block diagram of a lossless compression scheme.



(O-425) **Fig. 9.4.3 : Block diagram of lossless data compression**

Operation :

- The source data (data to be compressed) is applied to the **lossless encoder,** which consists of a predictor and an entropy encoder.
- The compressed data size is always smaller than the source data without any loss of information.
- The lossless compression technique generally accepts data to be a sequence of pixels in row. The processing of every pixel has two independent operations.
- The first operation creates prediction as to the numeric value of the next pixel. Predictors support linear combination of neighbouring pixel values.
- In the second operation, the difference between the predicted pixel value and real intensity of the next pixel is being coded by using entropy coder.
- The lossless compression is very important for the **text compression.**

9.4.4 Importance of Lossless Compression :

- It is very essential that the reconstruction of compressed text is exactly same to the original text because very small differences may result in words with very different meanings. For example, "Do not send money" and "Do now send money".
- A similar case stands true for computer files and for specific types of data such as bank records.
- If data of any type are to be processed or "enhanced" later to give more data, it is essential, that the integrity be saved.
- For example : Suppose a radiological image is compressed in lossy manner and the difference between reconstructed image Y and original image X was visibly undetectable.
- If this image was later enhanced, the previously undetectable differences may produce the artifacts. These artifacts can seriously misguide the radiologist.
- As the cost of this type of accident may be a human life, we should be very careful about using a compression technique, which produces a reconstruction, which is different from the original.

9.4.5 Advantages of Lossless Compression :

1. There is no data loss.
2. There is no loss of resolution.
3. Higher bit rate and sampling rate can enhance the quality by capturing very minute details. (Quality is preserved).
4. The size of compressed data is always smaller than the source data.
5. It is applied to graphic files, any type of computer data, text documents and software applications. (e.g. zip format).

9.4.6 Disadvantages :

1. It takes up more data space.
2. It uses more processing power when recording.
3. Files can not be transmitted quickly.
4. It does not reduce the file size as much as the lossy compression does.

9.4.7 Applications :

1. For the compression of graphic data.
2. For the compression of any type of computer data.
3. For text documents.

Note :	Lossless compression is best suitable for all the text documents containing important information such as bank statements.

9.5 Lossy Compression :

Definition :

- This is the class of data compression methods, that use inexact approximations and partial data discarding to represent the content.

Thus in this type of compression, there is a loss of information in a controlled manner, without a significant loss of important information.

The lossy compression is therefore not completely **reversible.** That means we cannot reproduce the exact replica of the original picture at the receiver.

But the advantage of this type is **higher compression ratios** than the lossless compression.

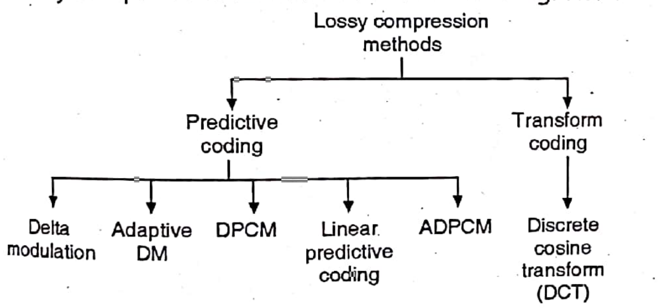
The lossless compression is used for the digital data. The lampel - Ziv algorithm is used for the same.

Applications :

- For many applications, the lossy compression is preferred due to its higher compression without a significant loss of important information.
- Lossy compression can be applied to graphical images and digitized voice.
- By their very nature, these digitized representations of analog phenomenon are not perfect and so a small loss of information due to compression does not matter much.
- Lossy compression is preferred for compressing the audio and video contents, where a slight loss of information does not affect the quality much.
- There is limit on the amount of compression in lossless compression methods.
- In the following subsections we will discuss some lossy compression methods.

Lossy compression methods :

- Lossy compression methods are as shown in Fig. 9.5.1.



(G-2264) **Fig. 9.5.1 : Lossy compression methods**

Predictive coding :

- The predictive coding is used to digitize an analog signal.
- In the digital systems such as PCM, each sample is quantized and encoded.
- However, in **predictive coding** we use the similarity in the adjacent samples.
- So, instead of quantizing and encoding each sample, the difference between the adjacent samples is quantized and encoded.
- This will reduce the number of bits required for encoding.
- Some of the examples of predictive coding are :
  1. Delta modulation
  2. Adaptive delta modulation
  3. Differential PCM (DPCM)
  4. Adaptive DPCM (ADPCM)

9.5.1 Principle of Lossy Compression :

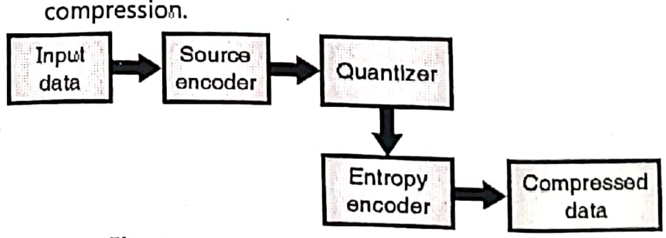
- It is possible to compress many types of digital data for reducing its size for storage and bandwidth to transmit it.
- The basic **information theory** says that there is an absolute limit in reducing the size of this data.
- When the data is compressed, its **entropy** increases. But entropy cannot increase infinitely.
- The compressed ZIP file is smaller than the original file, but it is not possible to compress it further. The point is that there is a limit on how much compression we can use.
- The files or data streams sometimes contain more information than needed. For example a picture may have more details than an eye can distinguish when it is reproduced.
- Or an audio file may contain very fine details which may not be identified in a loud passage.
- Lossy compression can be used, in such cases. It reduces the data without noticeable degradation in the quality of the reproduced picture or sound.

9.5.2 Compression Ratios :

- Higher the compression ratio, higher is the compression of data. Video can be compressed upto 100 : 1 with a very little loss of quality.
- Audio can be compressed upto a ratio 10 : 1 without much loss of quality.
- Still images are compressed at 10 : 1, but the loss of quality is more noticeable, especially on a closer inspection.

9.5.3 Block Diagram of Lossy Compression :

- Fig. 9.5.2 shows the simplified block diagram of lossy compression.



(O-428) **Fig. 9.5.2 : Block diagram of lossy compression**

- The input data which is to be compressed, is applied to a source encoder.
- The encoded signal is applied to a "quantizer" which approximates it to its nearest predecided value.
- An **entropy coding** is applied to the quantized output to obtain the compressed data.

9.5.4 Advantages of Lossy Compression :

1. We get far better compression ratio as compared to the lossless compression.
2. It significantly reduces the size of a file.
3. Due to lossy compression, the size of the storage is reduced significantly.
4. The transfer of content takes place quickly (faster data transfer).
5. Less bandwidth is required for the transmission of streamed video / audio.
6. The quality of audio / video does not degrade significantly.

9.5.5 Disadvantages :

1. There is some loss of quality of the reproduced video / audio contents.
2. Lossy compression can not be used for text data such as bank statements.
3. The original can never be reproduced.

9.5.6 Applications of Lossy Compression :

- Lossy compression, is most commonly used to compress the **multimedia data** such as audio, video and images.
- It is preferred in the applications such as **streaming media** and **Internet telephony.**

9.5.7 Comparison of Lossy and Lossless Compression :

MU : May 18

University Questions

Q. 1 Compare lossy with lossless data compression technique. **(May 18, 5 Marks)**

Sr. No.	Parameter	Lossless compression	Lossy compression
1.	Principle	There is no loss of information. Only the redundand bits are removed	There is a controlled loss of information
2.	Data integrity	Protected	Not protected
3.	Compression ratio	Small (less compression)	Large (More compression)
4.	Examples	RLE, Huffman coding, Lempel-Ziv coding	MP-3, JPEG, MPEG
5.	Reconstruction of original data	Possible	Not possible
6.	Quality of reproduced content	No degradation	Degradation takes place
7.	Suitable for	Text file compression	Compression of audio, images and video content

9.6 Optimal Compression :

- If we try to transmit the uncompressed speech or image over a digital network, it requires a large bandwidth.
- The uncompressed speech, video, images etc. need a large memory space for its storage.
- Therefore, it is necessary to compress audio, video or data information in order to reduce the bandwidth requirement and storage space.

Requirements of compression algorithms :

- There are many compression algorithms available. The general requirements of any compression algorithm are as follows :
  1. It must have an adequately high compression ratio yet the quality of the sound, video or data should not be affected too much.
  2. The compression algorithm should be simple.
  3. The compression technique should not introduce any delay (ideally).

9.6.1 Optimal Compression Algorithms :

MU : May 14, May 15, May 16

University Questions

Q. 1 What is compression ? List different compression algorithms. **(May 14, May 16, 3 Marks)**

Q. 2 What is compression ? List different compression algorithm. Why adaptive Huffman coding is used ? **(May 15, 4 Marks)**

- The compression ratio is a parameter which indicates the effectiveness of a compression algorithm.

It is defined as follows :

$$\text{Compression ratio} = 1 - \frac{\text{Compressed size}}{\text{Uncompressed size}} \times 100\ \%$$

- Two very important and widely used compression algorithms are :
  1. Huffman coding
  2. Arithmetic coding
- Two important approaches to the data compression are :
  1. Statistical data compression
  2. Dictionary systems
- Examples of dictionary systems are : LZ77 and LZ78 algorithms.
- For the image compression purpose the widely used compression algorithms are : GIF and JPEG.

9.7 Huffman Coding : **MU : Dec. 15**

University Questions

Q. 1 Describe Huffman decoding procedure with example. (Dec. 15, 10 Marks)

Concept :

- The Huffman code is a **source code.** It is a prefix code as well.
- Here word length of the code word approaches the fundamental limit set by the entropy of discrete memoryless source.
- This code is **"optimum"** as it provides the smallest average code word length for a given discrete memoryless source.
- It encodes each message transmitted by a DMS with different value of number of bits based on their probabilities.
- The Huffman code is used when the probability model for the source is known.
- The Huffman procedure is based on two observations regarding optimum prefix codes as follows :
  1. In an optimum code, the symbols having higher probability of occurance will be encoded into shorter codes than those having less probability of occurance.
  2. In an optimum code, the code length for the two symbols having the smallest probability will be same.

9.7.1 Minimum Variance Huffman Code :

Coding procedure :

- The coding procedure for the minimum variance Huffman code is as follows :
  1. The source symbols (messages) are arranged in the order of decreasing probability. The two source symbols having the lowest probability are assigned with binary digits 0 and 1.
  2. These two source symbols (messages) are then **"combined"** into a new source symbol (message) with probability equal to the sum of the two original probabilities. The probability of the new symbol is placed in the list of symbols as per its value.
  3. This procedure is repeated until we are left with only two source symbols (messages) for which a 0 and a 1 are assigned.
  4. The code of each original source symbol is obtained by working backward and tracing the sequence of 0s and 1s assigned to that symbol. (As shown in Ex. 9.7.1).

Huffman algorithm :

- The Huffman coding can be shown in the form of an algorithm as follows :

1. List source symbols (messages) in the order of decreasing probability.

↓

2. The two source symbols of lowest probability are assigned numbers 0 and 1.

↓

3. These two source symbols are combined into a new message.

↓

4. The probability of this new message is equal to the sum of probabilities of the two original symbols.

↓

5. The probability of this new message is placed in the list according to its value.

↓

6. Repeat this procedure until we are lift with only two source symbols, for which a 0 and a 1 are assigned.

(E-104)

Ex. 9.7.1 : Consider the five source symbols (messages) of a discrete memoryless source and their probabilities as shown in the Table P. 9.7.1(a). Follow the Huffman's algorithm to find the code words for each message. Also find the average code word length and the average information per message. **Dec. 15, 10 Marks, May 19, 5 Marks**

Table P. 9.7.1(a)

Message	m_1	m_2	m_3	m_4	m_5
Probability	0.4	0.2	0.2	0.1	0.1

Soln. :

(a) To find the code word for each message :

Steps to be followed :

Step 1 : Arrange the messages in the order of decreasing probabilities.

Step 2 : Assign numbers 0 and 1 to the two messages having lowest probability.

Step 3 : Combine these two messages into a new message and place this probability in the probability list as per its value.

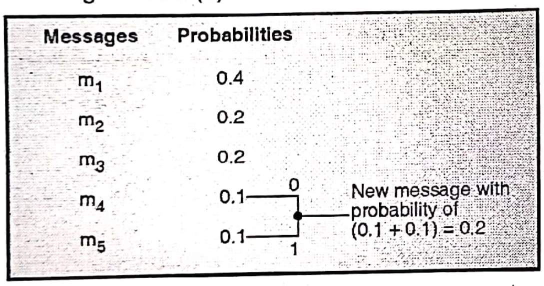
Steps 4 and 5 : Repeat this procedure.

Step 6 : Write the code word for each message by tracking back from the last stage to the first stage.

- Let us find code word for each message by following the Huffman's algorithm in steps :

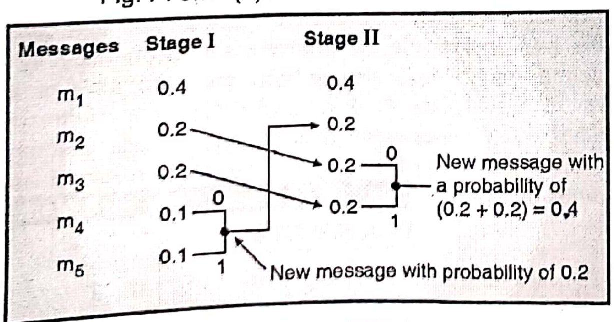
Step 1 : Arrange the given messages in the order of decreasing probabilities as shown in Fig. P. 9.7.1(a).

Step 2 : The two messages having lowest probability are assigned 0 and 1. The two messages with lowest probabilities are m_4 and m_5 as shown in Fig. P. 9.7.1(a).



(E-105) Fig. P. 9.7.1(a)

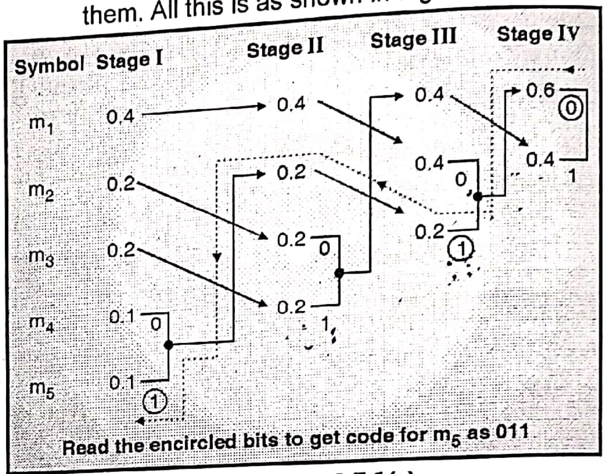
Step 3 : Now consider that these two messages m_4 and m_5 as being combined into a new message (Fig. P. 9.7.1(a)) and place the probability of the new combined message, in the list according to its value. Place the combined message as high as possible when its probability is equal to that of the other messages. This is as shown in Fig. P. 9.7.1(b).



(E-106) Fig. P. 9.7.1(b)

Step 4 : Now consider the two messages of lowest probabilities in stage II of the Fig. P. 9.7.1(b). Assign 0 and 1 to these two messages. Consider that these two messages are combined to form a new message with a probability of (0.2 + 0.2) = 0.4. Place the probability of the combined message according to its value in stage III. Place it as high as possible if the other messages have the same probability. This is as shown in Fig. P. 9.7.1(b).

Step 5 : Follow the same procedure till only two messages remain and assign the 0 and 1 for them. All this is as shown in Fig. P. 9.7.1(c).



(E-107) Fig. P. 9.7.1(c)

Step 6 : How to write the code word for a message ?

Consider the dotted path shown in Fig. P. 9.7.1(c). To write the code for message m_5 this path is to be used. Start from stage IV and track back upto stage I along the dotted path. And write down the code word in terms of 0s and 1s starting from stage IV.

The code word for message m_5 is 0 1 1.

- Similarly write code words for the other messages as shown in Table P. 9.7.1(b).

Table P. 9.7.1(b)

Messages	m_1	m_2	m_3	m_4	m_5
Probabilities	0.4	0.2	0.2	0.1	0.1
Code words	00	10	11	010	011

(b) To find the average code word length :

- The average code word length is given as,

$$L = \sum_{k=1}^{5} p_k \text{ [Length of } m_k \text{ in bits]}$$

$$= (0.4 \times 2) + (0.2 \times 2) + (0.2 \times 2) + (0.1 \times 3) + (0.1 \times 3) = 2.2$$

(c) To find the entropy of the source :

- The entropy of the source is given as,

$$H = \sum_{k=1}^{5} p_k \log_2 (1/p_k)$$

$$H = 0.4 \log_2 (1/0.4) + 0.2 \log_2 (1/0.2) + 0.2 \log_2 (1/0.2) + 0.1 \log_2 (1/0.1) + 0.1 \log_2 (1/0.1)$$

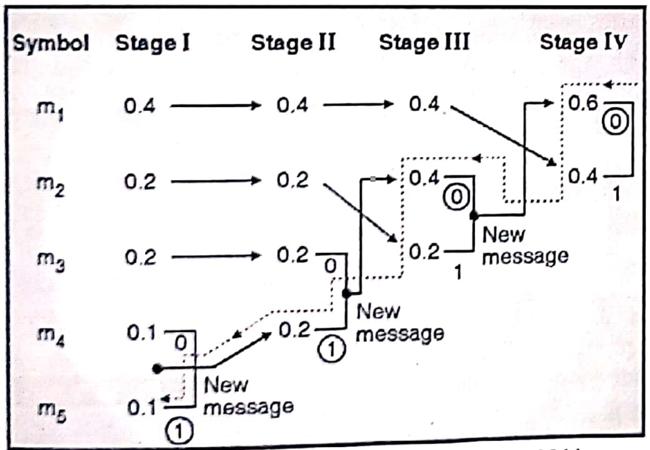
$$= 0.52877 + 0.46439 + 0.46439 + 0.33219 + 0.33219$$

$$\therefore H = 2.12193$$

Ex. 9.7.2 : Consider the same memoryless source as in Ex. 9.7.1, all the data is same. Find the Huffman code by moving the probability of the combined message as low as possible. Tracking backwards through the various steps find the code words of this second Huffman code.

Soln. :

- All the steps to be followed, are same as those followed for the Huffman's first code explained in Ex. 9.7.1, except for the change that the combined message is to be placed as low as possible.
- This is as shown in Fig. P. 9.7.2.



Read the encircled bits to get code for m_5 = 0011

(E-108) Fig. P. 9.7.2 : Huffman's second code

To find code word :

- This procedure is same as that followed in the previous example. Follow the dotted path in Fig. P. 9.7.2 to obtain the code for message m_5 as,
- Code word for the message m_5 is 0 0 1 1.
- Similarly the code words for the other messages can be obtained. They are as listed below :

Messages	Probability	Code word
m_1	0.4	1
m_2	0.2	0 1
m_3	0.2	0 0 0
m_4	0.1	0 0 1 0
m_5	0.1	0 0 1 1

- Note that to transmit the same messages as those of the previous example now we need more number of bits per message.

Ex. 9.7.3 : A discrete memoryless source has an alphabet of seven symbols with probabilities for its output as described in Table P. 9.7.3(a).

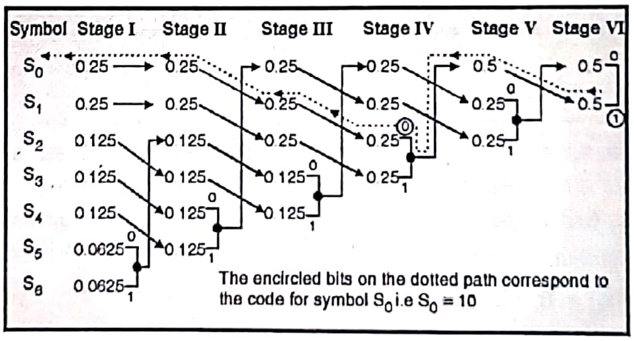
Table P. 9.7.3(a)

Symbol	S_0	S_1	S_2	S_3	S_4	S_5	S_6
Probability	0.25	0.25	0.125	0.125	0.125	0.0625	0.0625

Compute the Huffman code for this source moving the "combined" symbol as high as possible. Explain why the computed source code has an efficiency of 100 percent.

Soln.

- The Huffman code for the source alphabets is as shown in Fig. P. 9.7.3.



(E-109) Fig. P. 9.7.3 : Huffman code

- Follow the path indicated by the dotted line to obtain the code word for symbol S_0 as 10.
- Similarly we can obtain the code words for the remaining symbols.
- These are as listed in Table P. 9.7.3(b).

Table P. 9.7.3(b)

Symbol	Probability	Code word	Code word length
S_0	0.25	10	2 bit
S_1	0.25	11	2 bit
S_2	0.125	001	3 bit
S_3	0.125	010	3 bit
S_4	0.125	011	3 bit
S_5	0.0625	0000	4 bit
S_6	0.0625	0001	4 bit

- Each frame is divided into small grids called pixels. For the black and white TV, each 8 bit pixel represents one of 256 possible different gray levels, because $256 = 2^8$.
- For colour TV each pixel is represented by 24 bits out of which a group 8 bits is used to represent 8 bits one primary colour (Red, Green and Blue).
- The lowest resolution for a coloured picture is 1024 × 768 pixels per frame.
- So the number of bits corresponding to one picture is calculated as follows :

$$\begin{aligned} \text{Number of bits/pictures} &= 2 \times \text{Number of frames/sec.} \times \text{Resolution} \\ &= 2 \times 25 \times 1024 \times 768 \\ &= 944 \text{ Mbps} \\ &= 944 \times 10^6 \text{ bps} \end{aligned}$$

- To send video with minimum resolution, we need to transmit at 944 Mbps which requires a huge bandwidth, (approximately equal to 944/2 = 472 MHz) per channel.
- This much bandwidth is not available practically. So we have to use the compression technique so as to reduce the number of bits and hence to reduce the bandwidth.
- Compression is essential to send video over the Internet as well.
- It is possible to sample and quantize the video images so that a binary data stream can be used to represent the image in a satisfactory manner.
- It is interesting to note that we can represent a picture by anything from 1000 to 1 million bytes of data. The quality of picture will depend on the number of bytes used.
- For the storage and transmission of such a huge picture data, some kind of a compression is necessary.

9.13.3 JPEG Standards :

- Video is simply a sequence of images (plus sound). So video compression is called as image compression.
- One of the standards used for compression is JPEG i.e. Joint Photographic Experts Group standards. This is a standard for compressing still pictures such as photographs.

9.13.4 The JPEG Standard for Lossless Compression :

Lossless Options :

- The JPEG was formed jointly by CCITT and ISO, the two standard organizations.
- The lossless compression option of JPEG image compression standard is a description of 29 different coding systems for compressing images.
- So many approaches are described because the needs of different users are extremely different as compared to each others.
- In this section we will discuss only two methods which use the entropy coding.
- They are :
  1. Huffman code and
  2. Arithmetic code
- Out of these we have already discussed the Huffman code. Let us now discuss the **Arithmetic code.**

Arithmetic code :

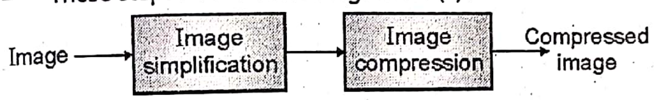
- This code achieves the compression in transmission or storage by making use of the probabilistic nature of the data.
- The number of bits after compression will be less than those used in the source data stream.
- Arithmetic code has an advantage over the Huffman code that it comes closer to the Shannon entropy limit of compression.
- The reason for this is that the Huffman code works effectively when the probabilities of the symbols can be expressed as fractions of powers of two. This constraint is not present for the arithmetic code.
- It is possible to achieve some compression if we can predict the next pixel in an image using the previous pixels.
- In this method one has to transmit only the difference in values (predicted and actual) which is called as the prediction coefficient. Note that the entire pixel is not to be transmitted.
- In JPEG, three are various schemes used for predicting the next pixel.
- These variations are not dependent on the users choice. But that scheme for prediction is selected which gives the best possible prediction.
- In all there are eight prediction methods available in JPEG coding standards one of them is not used for the lossless compression. The remaining seven are classified into the following categories :
  1. This predicts that the next pixel on the line has the same value as the previous pixel.
  2. This group of schemes predicts that the next pixel on a line has the same value as the pixel at the same position on the previous line.
  3. This group of schemes predicts that the next pixel on a line is related to a combination of the previous, above and previous to above pixel values. In one such technique they take the average of the three values.

- One of the lossless compression techniques used in JPEG standards is the **differential encoding.** It consists of the differences between the actual pixel value and the predicted value.
- Due to the redundancy present in an image, the difference between the predicted and actual pixel value will be small. So this scheme is suitable for the entropy coding.

9.13.5 The JPEG Standard for Lossy Compression :

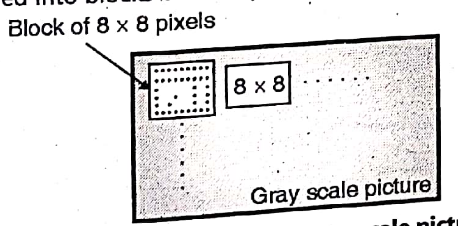
Concept :

- As some image distortion is acceptable to human eye, the JPEG standard includes a set of lossy compression options.
- The JPEG lossy compression first simplifies the image and removes the image complexity. Then the simplified image is passed through the lossless compression based on predictive filtering and Huffman or Arithmetic coding.
- These steps are shown in Fig. 9.13.3(a).



(O-554) Fig. 9.13.3(a)

- The lossy image simplification step called as the image reduction is carried out by using the Discrete Cosine Transform (DCT).
- If the picture is not coloured (gray scale), then each pixel can be represented by an 8-bit integer (i.e. 2^8 = 256 levels). If a coloured picture is to be represented, then we have to use 24 bits (3 × 8 bits) with each 8 bits representing one of the three basic colours (Red, Green and Blue).
- As shown in Fig. 9.13.3(b) in JPEG, a gray scale picture is divided into blocks of 8 × 8 pixels.

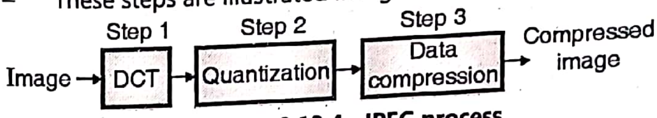


(E-1208) Fig. 9.13.3(b) : JPEG divides a gray sale picture into blocks of 8 × 8 pixels.

- The JPEG changes the picture into a linear (vector) set of numbers. This process is explained below.

Three steps in JPEG :

- The three steps in JPEG compression process are as follows :
  1. Discrete Cosine Transform (DCT).
  2. Quantization
  3. Data compression.
- These steps are illustrated in Fig. 9.13.4.



(E-1209) Fig. 9.13.4 : JPEG process

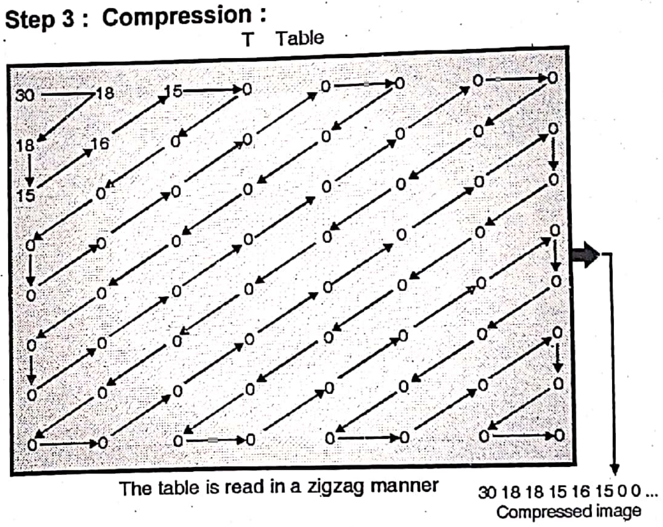
Step 1 : Discrete Cosine Transform (DCT) :

- The image in the form of blocks of 8 × 8 = 64 pixels is applied to the DCT box.
- Each 8 × 8 block will undergo a transformation called discrete cosine transform (DCT).

Step 2 : Quantization :

- The DCT transformed 8 × 8 blocks are then quantized. The process of quantization reduces the number of bits required to be used for encoding.
- Quantization is the process of approximation. Hence we lose some information and it is not possible to recover this information at all.
- This is why the JPEG is called as lossy compression.

Step 3 : Compression :



(E-1210) Fig. 9.13.5 : Compression

- In the first step (DCT) a transformation table T is created.
- After quantization, the values in this table are read to remove redundant zero bits.
- The table is read in a zig-zag manner, as shown in Fig. 9.13.5, to produce the compressed digital image.
- Note that the numbers in the table indicate the gray scale (shade in between black and white) and zeros indicate redundancy i.e. no change in shade.
- So at the output of the compressor, we get the compressed digitized image.

9.13.6 Discrete Cosine Transform (DCT) :

Definition :

- The discrete cosine transform is defined as follows :

$$Y(k, l) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} 4\,y(i, j) \cos\left[\frac{\pi k}{2N}(2i+1)\right] \cos\left[\frac{\pi l}{2M}(2j+1)\right]$$

- In this equation the image to be compressed is assumed to be of size N pixel by M pixel, y (i, j) is the intensity of the pixel in row i and column j.
- Y (k, l) is the DCT coefficient in the k^{th} row and l^{th} column of the DCT matrix.
- In the DCT all the multiplications are real. So there are less number of multiplications as compared to Discrete Fourier Transform (DFT).

A DCT representation f an image :

- For most of the images a large part of signal energy lies at the lower frequencies which appears in the upper left corner of the DCT as shown in Fig. 9.13.6.
- The values on the lower right side represent the higher frequencies as shown in Fig. 9.13.6 and these values are generally small.

DC coefficient

4.43	3.22	3.11	2.36
2.88	2.25	1.8	1.61
2.2	1.36	0.4	0.21
1.66	0.48	0.04	0.01

Low frequency coefficient

High frequency component

(E-1211) Fig. 9.13.6 : Typical values of DCT corresponding to an image

- The values of high frequency components are usually so small that we can neglect them without much distortion getting introduced.

Image reduction process :

- In JPEG, as mentioned earlier, the input image is divided into 8 × 8 pixel blocks, for application of DCT. For example an image of 256 × 256 pixel size, will be broken into 32 blocks each containing 8 × 8 = 64 pixels. Each of these 32 blocks are considered separately.
- The 64 pixel values in each block are then transformed by the DCT into a set of 64 new values. These new values are known as the **DCT coefficients** as shown in Fig. 9.13.7.
- The component on the top left in Fig. 9.13.6 is called as **DC coefficient** and its value is proportional to the average value of the 8 × 8 block of pixels. The remaining coefficients are called as the **AC coefficients.**
- As far as the image reduction using DCT is concerned, here is an interesting observation. For most images the maximum information lies in the low frequency spectrum than the high frequency components.
- So, in order to exercise the image reduction, we can represent the higher frequency components in an approximate manner or eliminate them totally. This will not have much of an effect on the quality of the image.This is how the DCT will achieve the image compression.

Steps followed for compression :

- The compression takes place by following the steps given below :
  1. The lowest (high frequency) coefficients are trimmed by setting them to zero. This will reduce the higher frequency components.
  2. The remaining low frequency higher value coefficients are then quantized. This will reduce some of the medium and low frequency components in the given image.
- The AC coefficients are quantized as shown in Figs. 9.13.7(a) and (b). This will convert most of the small value components to zero, as shown in Fig. 9.13.7(b).
- Finally the zigzag coding is used to obtain the code as shown in Fig. 9.13.7(b).

4.43	3.22	3.11	2.36
2.88	2.25	1.8	1.61
2.2	1.36	0.4	0.21
1.66	0.48	0.04	0.01

Quantization

4	3	3	2
3	2	2	2
2	1	0	0
2	0	0	0

Zigzag coding

4 3 3 3 2 2 2 1 2 2 2 0 0 0 0 0

(a) Original DCT coefficients **(b) Quantized DCT values and zigzag coding**

(E-1212) Fig. 9.13.7

- The zigzag coding is used in order to move smoothly from low frequency to high frequency components in the image without abrupt jumps.

9.13.7 Types of DCT :

- Types of DCT :
  1. One-dimensional DCT 2. Two-dimensional DCT
- In multimedia compression, we use two-dimensional DCT.

9.13.8 Limitations of JPEG Standard :

1. JPEG offers an excellent quality at high and mid bit rates. But at low bit rates the quality of JPEG is unacceptable (e.g. below 0.25 bits per pixel).
2. JPEG can not provide a superior performance at lossless and lossy compression.
3. The current JPEG standard provides some resynchronization markers, but the quality still degrades when bit errors are encountered.
4. JPEG was optimized for natural images. Therefore its performance on computer generated images and bi-level text images is poor.
5. Every next step in compressing the JPEG image degrades its quality.

9.13.9 Advantages of JPEG :

1. JPEG is extremely portable.
2. JPEG is compatible with almost every image processing applications.
3. It is easy to print JPEG images.
4. JPEG images can be stored quickly from a camera to storage device.
5. Size of JPEG images can be reduced and compressed which makes this file format suitable for transferring images over the internet because it consumes less bandwidth.
6. JPEG image can be compressed down to 5% of is original size.

9.13.10 Applications of JPEG :

1. Digital photography.
2. Internet imaging.
3. Image and video editing.
4. Security video cameras.
5. Medical image compression.
6. High speed video capture.

9.14 Graphics Interchange Format (GIF) :

- GIF is a bitmap image format introduced by Compuserve in 1987.
- It makes use of lossless compression algorithm.
- GIF supports upto 8 bits per pixel. This gives limitation upto 256 colors. The different colors are selected from 24 bit RGB color space.
- Most commonly, GIF is used for graphics or logos.
- A lossless compression of GIF images is done using LZW lossless data compression technique.
- The compression ratio of 2 : 1 is used.
- That means, compressed image is one half of the original image.
- Since maximum 256 color levels can be obtained; GIF is basically used for graphic and text based images.
- When interlaced GIF is used a low resolution image is displayed inside the webpage.
- This image gradually comes into focus.
- But if interlaced GIF is used then file size increases.
- When non-interlaced GIF is used then an empty box with red color "X" is displayed on the screen; until the full image is downloaded.
- Since human being is more sensitive to the sharpness of edges; GIF image appears more sharper than JPEG.
- Generally two types of GIF are used :
  1. Animated GIF
  2. Transparent GIF
- In animated GIF, multiple images are stored and they are controlled in such a way that appearance is like a real time animation.
- In case of transparent GIF is a special color is used as a background, so image appears as transparent.
- GIF uses a small indexed table (palette) of colours which contains 256 (2^8), colours. GIF maps a true colour with the indexed table colour.

9.14.1 Advantages :

- It is simple bit map image format which has wide support and portability.
- GIF supports transparency, animation and interlacing.
- Lossless compression is achieved by using fewer colors.
- GIF images are much sharper than JPEG.

9.14.2 Disadvantages :

- Since only 256 colors for each frame can be used as GIF is not applicable for reproducing color photographs and images having continuous colors.
- LZW compression algorithm is used, but this algorithm is not designed specifically for graphics.
- Due to this, GIF is not suitable for black and white images and true color images.