Compression and Decompression

What is Multimedia?

- Accessing information involves various media forms
- Media is some form of information like text, audio, video, etc.
- A web page primary contains text information, but it also can have pictures, animations, video clippings, audio commentaries and so on.
- When there is more than one form of media present, we call them as multimedia.
- Examples: Television(audio, video)
- A multimedia signal is one that integrates signals from several media sources, such as video, audio, graphics, animation, text in a meaningful way to convey some information.

Elements of a multimedia system

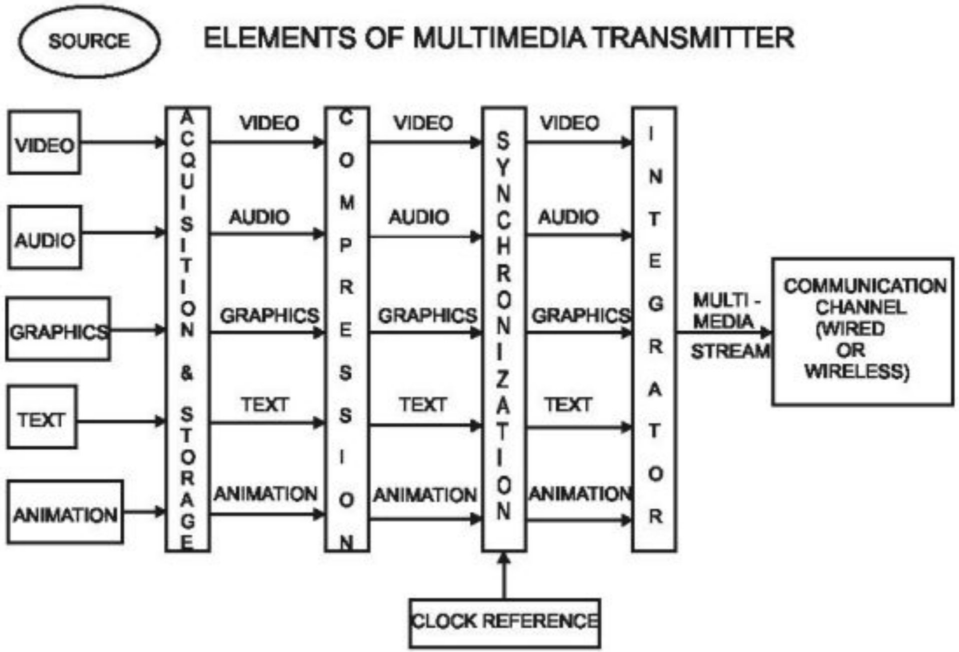


Fig 1.1: *Elements of Multimedia Transmitter*

Elements of a multimedia system

- Data acquisition and storage - get data
- Compression - make data smaller and eliminate inherent redundancies present in the media streams
- Synchronization - synchronize multiple types of media by insertion of time-stamps
- Integrator - integrates the individual streams from all media sources and transmits it through multimedia stream

On the receiver do the literal opposite.

Elements of a multimedia system - Receiver

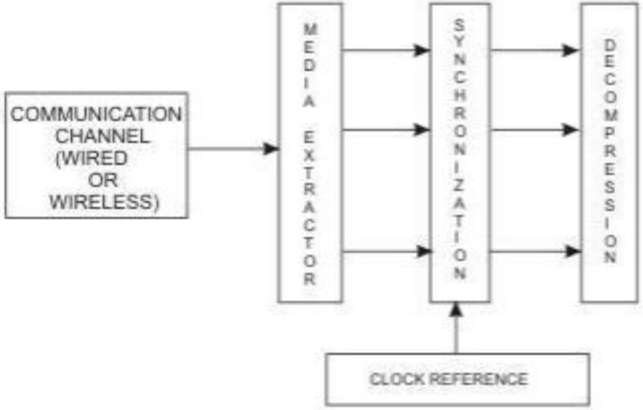


Fig 1.2 Elements of multimedia Receiver

Challenges involved with multimedia communication

- Bandwidth limitations of communication channels.
- Real-time processing requirements.
- Inter-media synchronization.
- Intra-media continuity.
- End-to-end delays and delay jitters.
- Multimedia indexing and retrieval.

- Real time processing requirements:
    - Data compression is a crucial technique for multimedia content, but processing time is a critical factor.
    - If processing time is too long, the benefits of compression can be lost
    - For instance, in still image compression, a 20:1 ratio can transmit an image in a minute, but slow processing could negate the advantage.
- Inter-media synchronization:
    - The media streams are available from different and independent sources and are asynchronous with respect to each other
    - Multimedia standards employ "time-stamping" with a system clock reference (SCR) to ensure proper synchronization. Time-stamps are appended to audio, video, and other media packets before integration.

→ Intra-media continuity:

- The extent of data compression with acceptable reconstruction quality is highly data-dependent. Wherever redundancy is more, high compression ratios are achievable, but redundancy may vary. Due to this we get variable bit rates, especially in video sequences.
- To address this, a buffer is used to accommodate variable bit rate sources and maintain a constant bit rate for transmission.
- Buffer management is crucial to avoid underflow (channel has no data) or overflow (buffer full so data discarded) issues, which can disrupt continuity during presentation.

→ End-to-end delays and delay jitters:

- In a multimedia broadcast or multimedia conferencing, if the users receive the multimedia contents after considerable delays or different users receive the same contents at different times, the interactivity is lost.
- The multimedia standards available till date have addressed this problem and specified what is acceptable.

➔ Multimedia Indexing and retrieval:

- **Growing Multimedia Data:** Digital storage media prices decrease, while storage capacity increases, leading to a increase in multimedia file availability.
- **Challenges in File Retrieval:** Managing a large number of multimedia files without proper indexing can make retrieval difficult due to search complexities.
- **Content-Based Query Systems:** Efficient retrieval can be achieved by organizing multimedia files based on their content and implementing content-based query systems.
- **Video Summaries:** Quick browsing of multimedia files often requires video summaries, but creating these summaries is a challenging task.

Redundancies

- The biggest challenge in multimedia communication is to transmit the multimedia signals, especially the image and the video signals through limited bandwidth channels.
- So we need to compress this data in order to send it across quicker.
- We rely on the redundancies in the data to compress the data.
- 2 types of redundancies:
  - Statistical Redundancy
  - Psychovisual Redundancy

Statistical Redundancy

- **Statistical Redundancy**: Occurs in images because neighboring pixels tend to have similar intensities, except at object boundaries or illumination changes.
- **Spatial Redundancy in Still Images:** In still images, statistical redundancies are primarily spatial, present along both the x and y dimensions, due to the nature of natural two-dimensional images.
- **Temporal Redundancy in Video:** Video signals exhibit temporal redundancy as intensities of the same pixel positions across successive frames are often very similar, except in the presence of substantial motion.

Psychovisual Redundancy

- **Psychovisual Redundancy:** Arises from human perception capabilities.
- **Perception vs. Detail:** Our eyes are more sensitive to slow illumination changes than fine details and rapid intensity changes.
- **Balancing Preservation and Quality:** Exploiting psychovisual redundancy involves determining the extent to which we should preserve image details for perception and where we can compromise on reconstructed image quality.

Types of compression

- Lossless image compression
- Lossy image compression

Lossless compression

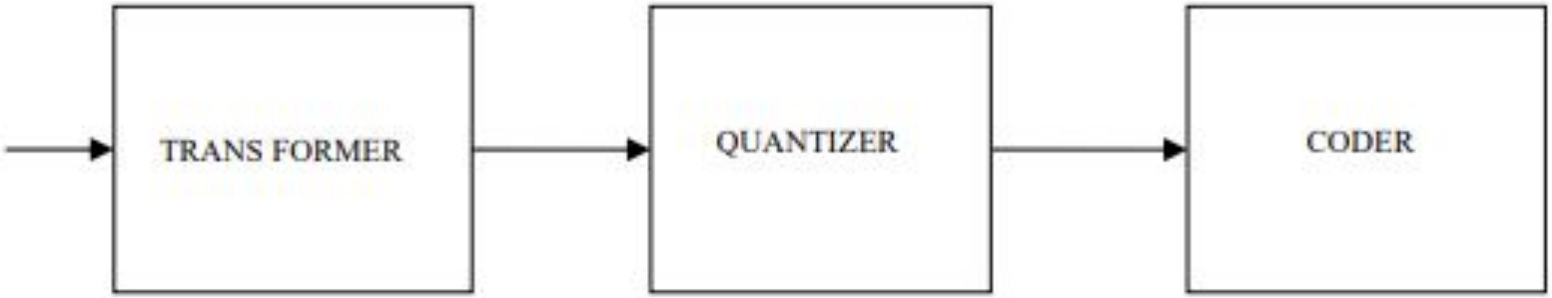
- **Lossless Image Compression:** Does not result in data loss, ensuring that the data stream before encoding and after decoding is identical, preserving image quality.
- **Statistical Redundancy Exploitation:** Lossless compression relies on statistical redundancy within the data.
- **Symbol Coding:** By transforming the image into symbols and assigning shorter code words to frequently occurring symbols and longer code words to less frequent symbols, compression is achieved.
- **Reversible Process:** The encoding and decoding process in lossless compression is fully reversible due to one-to-one mapping between symbols and their codes.
- **Limited Bandwidth Reduction:** Lossless compression achieves only modest bandwidth reduction for data transmission but maintains image quality without distortion.

Lossy compression

- **Lossy Image Compression:** Involves data loss, resulting in a reduction in the quality of image reconstruction.
- **Symbol Transformation:** Similar to lossless compression, lossy compression transforms the image into symbols.
- **Quantization:** Symbols are mapped to a discrete set of allowable levels, which leads to data compression.
- **Irreversible Mapping:** Quantization, a many-to-one mapping, is irreversible, and exact reconstruction is not possible.
- **Acceptable Loss:** Lossy compression is acceptable when the reduction in reconstruction quality is tolerable to human visual perception, and it allows for significant compression.
- **Exploiting Psychovisual Redundancy:** Lossy compression schemes rely on psychovisual redundancy and aim to identify areas where quality loss can be tolerated.

Lossless compression	Lossy compression
There is no loss of data	There is always a loss of data
Exactly reversible.	Not reversible.
Exploits statistical redundancy	Exploits psychovisual redundancy.

Elements of image compression system



➔ Transformer:

- **Transformer Block:** Part of image compression(no compression happens here), it transforms input data to facilitate compression.
- **Local vs. Global Transformation:** The transformation can be local (involving nearby pixels) or global (involving the entire image or pixel blocks).

➔ Quantizer:

- **Quantizer block:** It generates a limited number of symbols for representing the transformed signal.
- **Many-to-One Mapping:** Quantization is a many-to-one mapping, meaning it's an irreversible process.
- **Two types:**
  - Scalar quantization, which quantizes data element by element
  - Vector quantization, which quantizes a block of data at once.
- **Lossy Block:** Quantization is the only lossy block in the image compression system, as it introduces some loss of data.

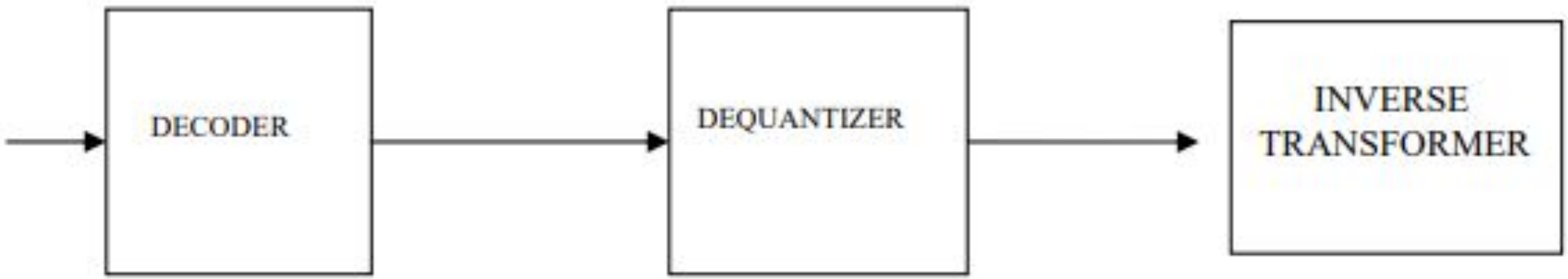
➔ Coder:

- **Coding Symbols:** Coders assign code words to the symbols obtained after quantization.

2 types:

- **Fixed-length coding (FLC):** which have codeword length fixed, irrespective of the probabilities of occurrence of quantized symbols.
- **Variable length coding (VLC):** also known as entropy coding, assigns code words in such a way as to minimize the average length of the binary representation of the symbols. This is achieved by assigning shorter code words to the more probable symbols.

Elements of Image decompression system



Literally the opposite

Quantizer

- Quantization is the process of mapping a set of continuous-valued samples into a smaller, finite number of output levels.
- 2 types: scalar, and vector quantization
- In **scalar quantization**, each sample is quantized independently.

A scalar quantizer Q(.) is a function that maps a continuous-valued variable s having a probability density function p(s) into a discrete set of reconstruction levels r_i (i=1,2,....,L) by applying a set of the decision levels d_i (i=1,2,....,L)

$Q(s) = r_i \quad \text{if } s \in (d_{i-1}, d_i] \quad i=1,2,...,L$

L is the number of levels.

- In **vector quantization**, each of the samples is not quantized. Instead, a set of continuous-valued samples, expressed collectively as a vector is represented by a limited number of vector states.

  Let $\dot{s} = Q(s)$ be quantized variable, then error $e = s - \dot{s}$, and distortion D is measured as MSE(Mean square error) $D = E(s - \dot{s})^2$

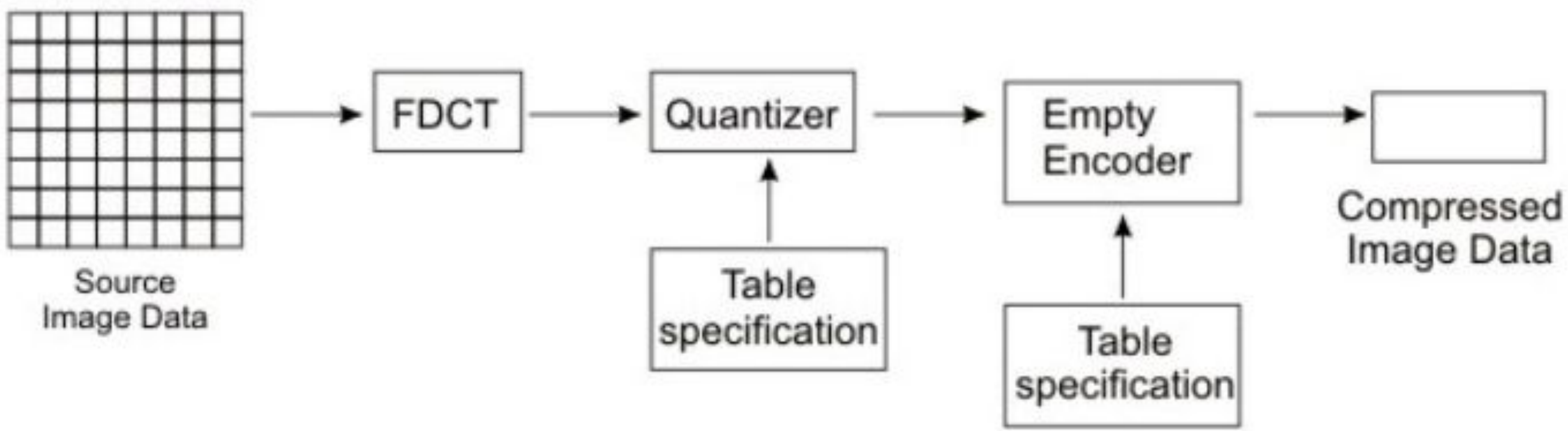
  Now choose d_i and r_i such that D is minimized.

Still Image Compression Standards - JPEG

Introduction

- **Importance of Coding Standards:** In the rapidly evolving field of imaging technology and image compression, coding standards are essential.
- **Compatibility and Interoperability:** Coding standards ensure compatibility and interoperability between image communication and storage products from different vendors.
- **No Standards, No Communication!!!! :** Without standards, encoders and decoders cannot effectively communicate, leading to challenges in data exchange and service provision.
- **JPEG Standard:** The JPEG standard is the most widely adopted and used standard for compressing and coding continuous tone monochrome and color images of various sizes and sampling rates.

Architecture

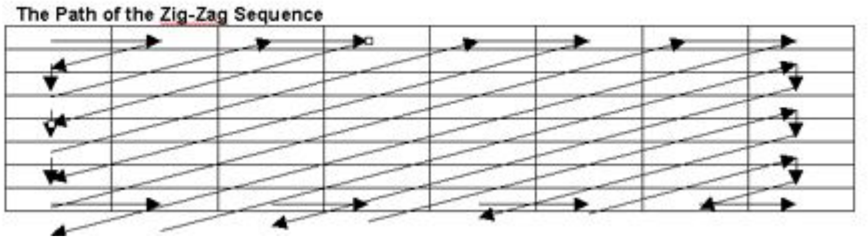


→ Forward Discrete Cosine Transform (FDCT):

- The still images are first partitioned into non-overlapping blocks of size 8x8
- The image samples are shifted from unsigned integers with range [0, 2^{p-1}] to signed integers with range [-2^{p-1}, 2^{p-1}], where p is number of bits(8 here)
- DCT algorithm transforms spatial data into frequency components using cosine functions

➜ **Quantization:**

- Each of the 64 coefficients from the FDCT outputs of a block is uniformly quantized according to a quantization table.
- Since the aim is to compress the images without visible artifacts, each step-size should be chosen as the perceptual threshold or for “just noticeable distortion”.
- The quantized coefficients are zig-zag scanned, this is done to exploit redundancy



➔ Encoder:

- Then we encode DC and AC.
- DC
  - **DC represents overall brightness**: The DC coefficient reflects the average brightness of a block in an image.
  - **Encoding DC**: To save space, the DC coefficient is encoded as the difference between its value and the DC coefficient of the previous block. This process uses less data because it focuses on how much the brightness changes from one block to the next.
  - **Example:** If the DC coefficient of one block is 100 and the next block's DC coefficient is 105, instead of storing 105, you encode it as +5 (indicating an increase of 5).
- AC
  - **AC represents image details**: AC coefficients encode the fine details within an image block.
  - **Encoding AC**: AC coefficients are grouped into pairs, each pair consisting of (run, level). The "run" indicates how many consecutive zero coefficients came before the non-zero coefficient. The "level" represents the actual value of the non-zero coefficient.
  - **Run-Length Encoding**: This method efficiently compresses sequences with long runs of zeros, and it allows you to represent both zero and non-zero values compactly.
  - **Example**: Instead of storing a long string of zeros, you can encode it as (5 zeros, 10), meaning there are 5 consecutive zeros followed by a non-zero value of 10.

→ **Entropy Coder:**

- JPEG standard specifies two methods - Huffman and arithmetic coding.
- **Baseline sequential JPEG**: Huffman coding is used exclusively.
- Huffman coding requires that one or more sets of coding tables are specified by the application. The same table used for compression is used needed to decompress it.
- The baseline JPEG uses only two sets of Huffman tables – one for DC and the other for AC

Modes of operation in JPEG

- Baseline or sequential encoding
- Progressive encoding
  - Progressive scanning through spectral selection
  - Progressive scanning through successive approximation
- Hierarchical encoding
- Lossless encoding

Baseline encoding

- Baseline sequential coding is designed for images with 8-bit samples.
- Utilizes Huffman coding exclusively for entropy encoding.
- Each image block is encoded in a single left-to-right and top-to-bottom scan.
- Encodes and decodes 8x8 blocks with full precision one at a time.
- Supports interleaving of color components
- The sequence includes FDCT, quantization, DC difference calculation, and zig-zag ordering.
- Products claiming JPEG compatibility must include support for at least the baseline encoding system.

Progressive encoding

- Each block is encoded in multiple scans.
- Each scan follows zig-zag ordering, quantization, and entropy coding similar to baseline encoding, but takes much less time to encode and decode.
- Each scan contains only part of the complete information, leading to faster encoding and decoding.
- With the first scan, a basic image can be reconstructed, and successive scans refine image quality
- **Example:** web page image loading: you initially see a lower-quality image, but it progressively improves.

Progressive scanning through spectral selection

- **Initial Low-Frequency Scan:** The first scan transmits specific low-frequency DCT coefficients within each block.
- The image reconstructed at the decoder from the first scan appears blurred due to the absence of high-frequency details.
- Subsequent scans encode bands of coefficients with higher frequencies than the previous scan and so with each scan the image is enhanced further.

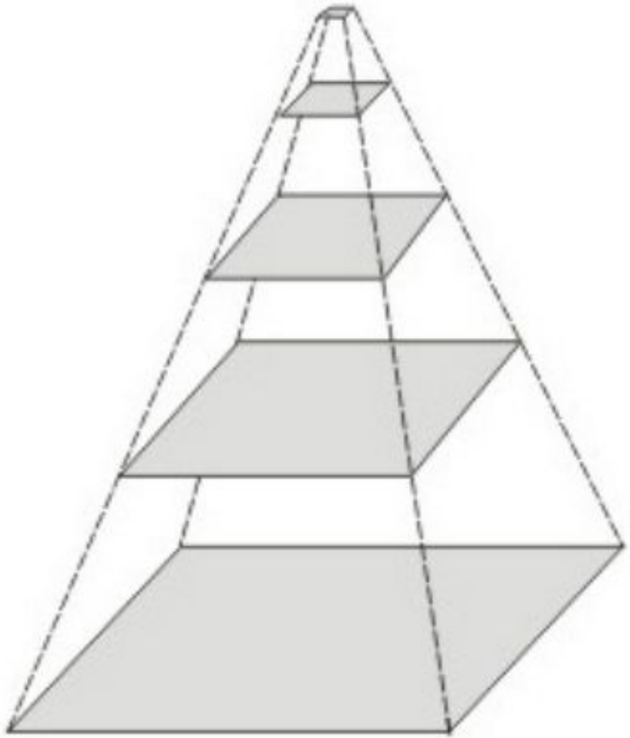
Progressive scanning through successive approximation

- Each scan encodes all coefficients within a block, but not to their full quantized accuracy.
- In the first scan, only the N most significant bits of each coefficient are encoded (N is specifiable).
- Successive scans add lower significant bits of coefficients until all bits are sent.
- The resulting reconstruction quality is good even from the early scans, as the high frequency coefficients are present from the initial scans.

Hierarchical encoding

- Pyramidal Structure: Image to be encoded is organized into a pyramidal structure with multiple resolutions.
- Layer Arrangement: Original image (finest resolution) is at the lowermost layer, and reduced resolution images are on upper layers.
- Resolution Reduction: Each layer decreases resolution with respect to the adjacent lower layer by a factor of two, in the horizontal, vertical, or both directions.
- Special Case of Progressive Encoding: Hierarchical encoding is akin to progressive encoding but with increasing spatial resolution between stages.

Hierarchical encoding

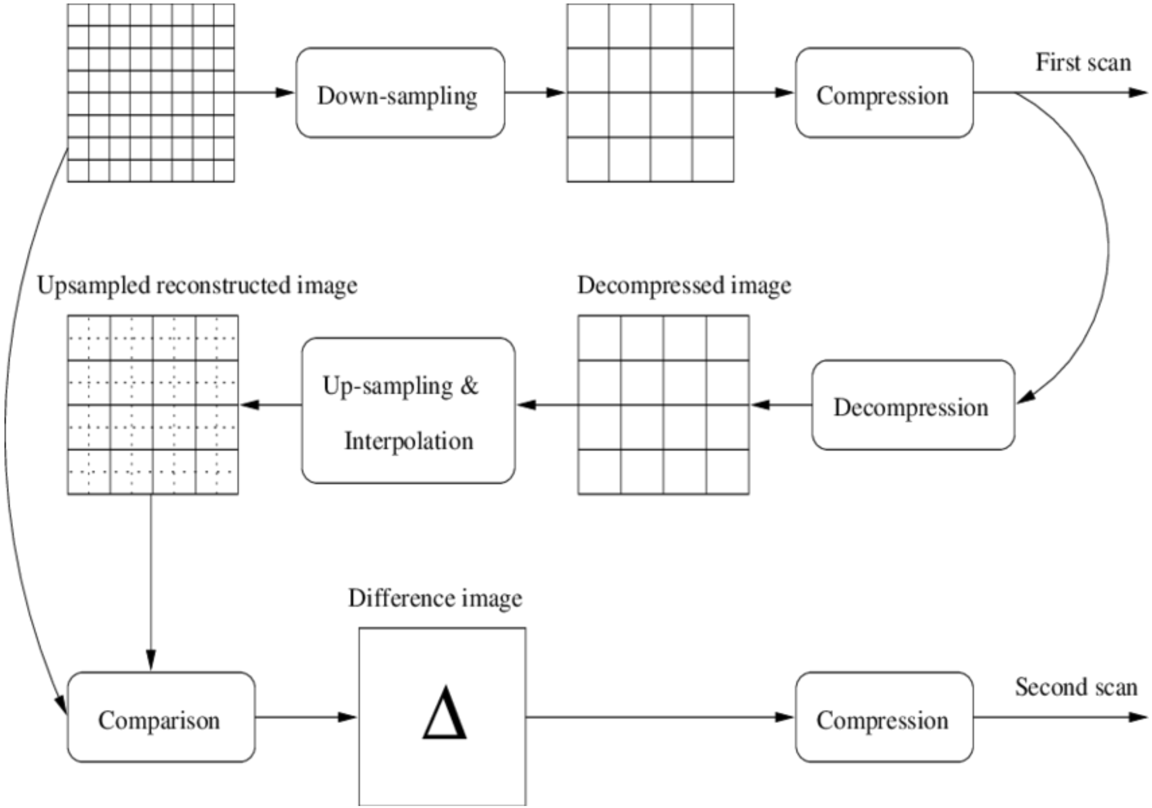


Hierarchical encoding

Steps Involved in Hierarchical Encoding:

1. **Resolution Reduction**: Obtain reduced resolution images starting with the original. Reduce resolution by a factor of two (horizontally, vertically, or both).
2. **Topmost Layer Encoding**: Encode the reduced resolution image from the topmost (coarsest) layer using baseline, progressive, or lossless encoding.
3. **Decode and Up-sample**: Decode the above image. Interpolate and up-sample it by a factor of two horizontally and/or vertically. Use this interpolated image as a prediction for encoding the next lower layer (finer resolution).
4. **Difference Encoding**: Encode the difference between the next lower layer and the predicted image using baseline, progressive, or lossless encoding.
5. **Repeat Stages**: Repeat the encoding and decoding steps until the lowermost layer (finest resolution) is reached.

Hierarchical encoding

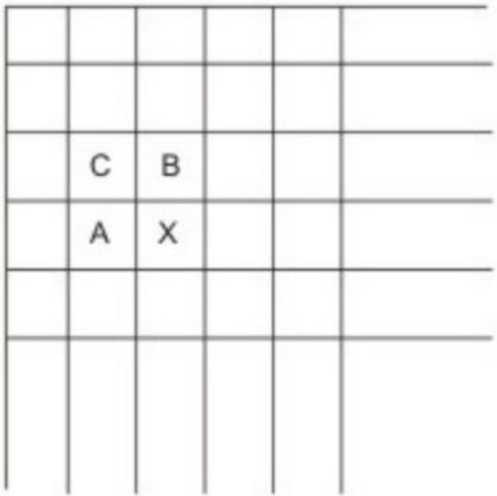


Lossless encoding

- **Predictive Coding**: Lossless encoding in JPEG employs a simple predictive coding mechanism. It doesn't use the FDCT + Entropy coder for encoding or the Entropy decoder + IDCT for decoding.
- **Quantization Elimination**: Unlike lossy encoding, which involves quantization, lossless encoding eliminates the quantization step to preserve all image details.
- While lossless encoding doesn't achieve as high compression ratios as lossy encoding, it's suitable for scenarios where preserving every bit of image data is essential.

- **Predictive Coding Mechanism**: Instead of using the 8x8 block structure, lossless encoding predicts each pixel based on three adjacent pixels. The prediction is based on one of eight possible predictor modes. An entropy encoder is then used to encode the predicted pixel obtained from the lossless encoder

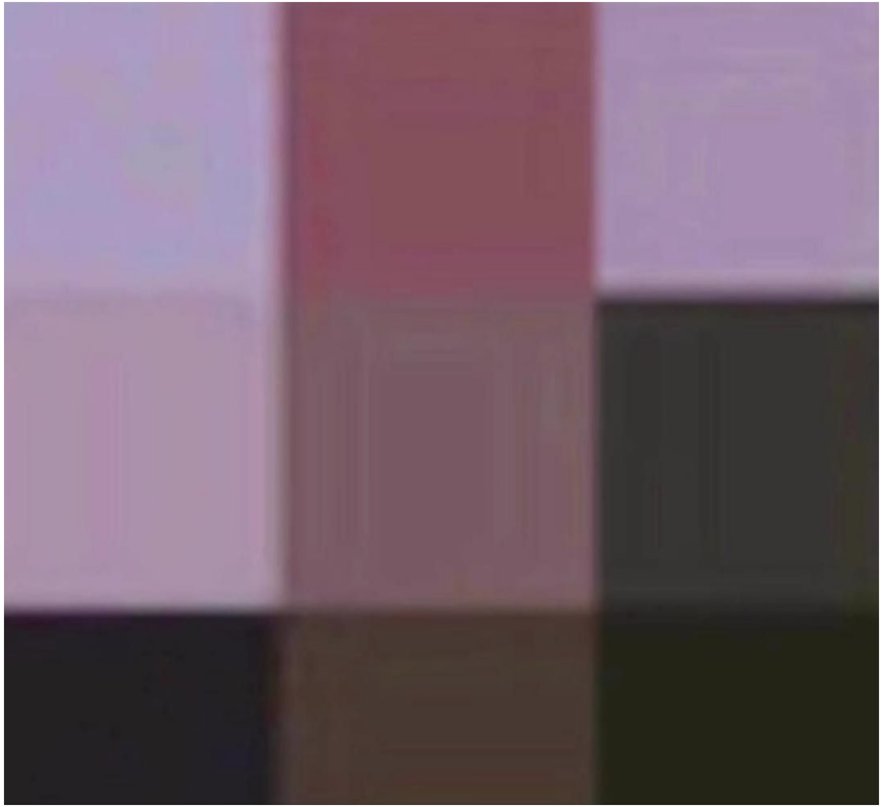
Selection Value	Prediction
0	None
1	A
2	B
3	C
4	A+B-C
5	A+(B-C)/2
6	B+(A-C)/2
7	(A+B)/2



Conclusion: Compression - before

Conclusion: Compression - after

Conclusion: Decompression - before



Conclusion: Decompression - after

References

- "Multimedia Systems Design" Prabhat K. Andleigh, Kiran Thakrar
- https://nptel.ac.in/courses/117105083
- https://cs.stanford.edu/people/eroberts/courses/soco/projects/data-compression/lossy/jpeg/lossless.htm
- https://www.researchgate.net/publication/2799814_Overview_of_Still-picture_and_VideoCompression_Standards
- https://www.youtube.com/watch?v=dQw4w9WgXcQ