Video Compression

Digital Video Fundamentals

- Digitization of video streams yields a sequence of *discrete* samples (called *frames* or *images*)
- Images consist of an array of *pixels* - the size of the array denotes the spatial resolution
- The radiant energy at each pixel is quantized to a linear space (e.g., the RGB color information associated with each pixel)

Need For Video Compression

- Large data rate and storage capacity requirement

Satellite imagery	180x180 km^2 30 m^2 resolution	600 MB/image
NTSC video	30 frames/s, 640x480 pixels, 3 bytes/pixel	30 MBytes/s

- Compression algorithms exploit:
  - Spatial redundancy (i.e., correlation between neighboring pixels)
  - Spectral redundancy (i.e., correlation between different frequency spectrum)
  - Temporal redundancy (i.e., correlation between successive frames)

Requirements for Compression Algorithms

- Objectives:
  - Minimize the complexity of the encoding and decoding process
  - Ensure a good quality of decoded images
  - Achieve high compression ratios
- Other general requirements:
  - Independence of specific size and frame rate
  - Support various data rates

Classification of Compression Algorithms

- Lossless compression:
  - Reconstructed image is mathematically equivalent to the original image (i.e., reconstruction is perfect)
  - Drawback: achieves only a modest level of compression (about a factor of 5)
- Lossy compression:
  - Reconstructed image demonstrates degradation in the quality of the image $\Rightarrow$ the techniques are irreversible
  - Advantage: achieves very high degree of compression (compression ratios up to 200)
  - Objective: maximize the degree of compression while maintaining the quality of the image to be "virtually lossless"

Compression Techniques: Fundamentals

- Entropy encoding:
  - Ignores semantics of input data and compresses media streams by regarding them as sequences of bits
  - Examples: run-length encoding, Huffman encoding, ...
- Source encoding:
  - Optimizes the compression ratio by considering media specific characteristics
  - Examples:
    - Predictive coding: e.g., DPCM
    - Layered coding: e.g., bit-plane coding, sub-sampling
    - Transform coding: e.g., DCT, FFT, Wavelet, ...
- Most compression algorithms employ a *hybrid* of the above techniques

Entropy Coding

- Run-length encoding:

$$a\ a\ a\ a\ a \rightarrow 5a$$

- Huffman encoding:
  - Employs variable length codes
  - More frequently occurring values are assigned fewer bits $\Rightarrow$ exploits the statistical distribution of the values within an image
  - Encoder and decoder must share the codebook
  - Example

Predictive Coding

- Basic technique:
  - Predict the value at a pixel by using the values of the neighboring pixels; and
  - Encode the difference between the actual value and the predicted value

$$e_m = x_m - \widehat{x_m}$$

- Predictor:
  - Dimension of the predictor
  - Order of the predictor: number of pixels used
  - Example of a third order predictor:

$$\widehat{x_m} = 0.75a - 0.5b + 0.75c$$

- Huffman encoding of differential images

Bit-plane Encoding

- A $N \times N$ image with k bits per pixel can be viewed as k - $N \times N$ bit planes
- Encode each bit plane separately
- Advantages:
  - Permits progressive transmission of encoded images (most significant bit plane first - since it generally contains more information)
- Encoding should be carried out such that separate encoding yields better performance than jointly encoding the bit planes
  - Gray codes are better suited as compared to binary encoding

Transform Coding

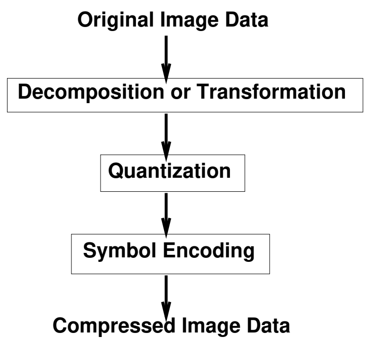
- Subdivide an individual $N \times N$ image into several $n \times n$ blocks
- Each $n \times n$ block undergoes a reversible transformation
- Basic approach:
  - De-correlate the original block $\Rightarrow$ radiant energy is redistributed amongst only a small number of *transform coefficients*
  - Discard many of the low energy coefficients (through quantization)
- General requirements:
  - Image independence
  - Should be computationally efficient

Transform Coding: An Example

- Consider a 2-point transform (consisting of adjacent pixels)
- Since adjacent pixels are typically highly correlated, most points lie along the 45^0 line defined by $x_1 = x_2$
- Variance in both x_1 and x_2 dimension is large $\Rightarrow$ encoding any one of them with the average value may lead to large error
- Solution: rotate axis by 45^0 $\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$
- Encode the data with $y_2 = 0$ (i.e., the mean value for y_2), resulting in the mean square error $= \sigma^2_{y_2}$, which is much smaller than either $\sigma^2_{x_1}$ or $\sigma^2_{x_2}$
- Reverse transformation: $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$

Lossy Compression Techniques

- Image decomposition or transformation:
  - Reduce dynamic range of signal, eliminate redundant information, or provide a representation that is more amenable to efficient coding
- Quantization:
  - Quantizer: staircase function that maps a continuum of input signal values into a finite number of output levels
  - Many-to-one mapping, reduces the number of possible output symbols at the cost of increased error in the reconstructed image
  - Can be considered a control knob that trades of image quality for bit rate
- Symbol encoding



JPEG Compression Standard

- Joint Photographic Experts Group: CCITT-ISO
- General purpose standard for:
  - Still images
  - Gray-scale or color images
  - Wide range of applications
- Modes of operation:
  - Sequential encoding: single left-to-right, top-to-bottom scan
  - Progressive encoding: multiple scans over the image
  - Lossless encoding: Uses predictive coding techniques
  - Hierarchical encoding: multiple resolution levels
- Transform coding in JPEG: Discrete Cosine Transform (DCT)-based
  - DCT is the one of the best known transforms, and is a close approximation to the optimal for a large class of images
  - Fast algorithm exists; relatively straightforward implementation

DCT in JPEG

- Partition the original image into 8×8 pixel blocks and independently apply DCT to each block
- Precise definition of FDCT (Forward DCT):

$$F(u,v) = \frac{1}{4}C(u)C(v) * \left[\sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) * \cos\frac{(2x+1)u\pi}{16} * \cos\frac{(2y+1)v\pi}{16} \right]$$

where

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } u, v = 0 \\ 1 & \text{otherwise} \end{cases}$$

- Output of DCT: one DC-coefficient and 63 AC-coefficient
- FDCT enhances compression by concentrating most of the energy in the signal in the lower spatial frequencies

DCT and JPEG Coding

- DCT in principle is lossless; merely transforms them into a domain in which they can be more efficiently encoded
- However, FDCT and IDCT require the computation of cosine functions $\Rightarrow$ physical limits on the resolution of numbers may introduce inaccuracies
- Lossless JPEG does not rely on DCT
- Many approximate algorithms for FDCT have been defined - JPEG does not specify a unique algorithm

Quantization in JPEG

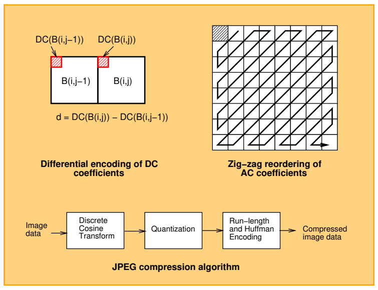
- Each of the 64 DCT coefficients is quantized, using a 64-element quantization table
- Main purpose: achieve high compression by representing DCT coefficients with no greater precision than necessary
- Many-to-one mapping $\Rightarrow$ fundamentally lossy
- Involves division of each DCT coefficient by the corresponding quantizer element, followed by rounding to the nearest integer

$$F^Q(u,v) = Integer\ \ Round\ \left(\frac{F(u,v)}{Q(u,v)}\right)$$

- Quantization table carefully determined based on source image characteristics, display characteristics, viewing distance, etc.

Encoding Quantized Images in JPEG

- Differential encoding of DC coefficients
- Zig-zag sequencing of AC coefficients
  - Facilitates entropy coding by placing high-energy, low-frequency coefficients before the low-energy, high-frequency coefficients
- Huffman coding used for both DC and AC coefficients



Department of Computer Science, UMass Amherst

MPEG

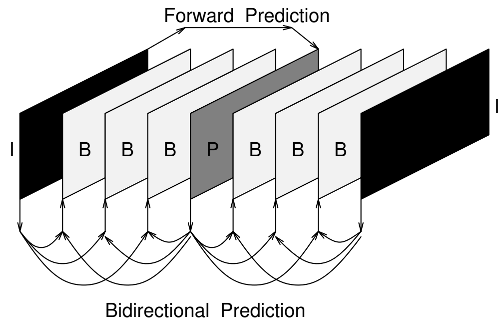
- Moving Picture Experts Group
- JPEG does not exploit temporal (i.e., frame-to-frame) redundancy present in all video sequences
- MPEG exploits temporal redundancy
- MPEG requirements:
  - Random access
  - Fast searches - both forward and reverse
  - Reverse playback
  - Audio-video synchronization
  - Robustness to errors
  - Low encoding/decoding delay
  - Editability

MPEG - An Overview

- Two categories: intra-frame and inter-frame encoding
- Contrasting requirements: delicate balance between intra- and inter-frame encoding
  - Need for high compression ⇒ only intra-frame encoding is not sufficient
  - Need for random access ⇒ best satisfied by intra-frame encoding
- Overview of the MPEG algorithm:
  - DCT-based compression for the reduction of spatial redundancy (similar to JPEG)
  - Block-based motion compensation for exploiting the temporal redundancy
    - Motion compensation using both *causal* (predictive coding) and *non-causal* (interpolative coding) predictors

Exploiting Temporal Redundancy

- Three types of frames in MPEG:
  - I-frames:
    - Intra-coded frames, provide access points for random access - yield moderate compression
  - P-frames:
    - Predicted frames are encoded with reference to a previous I or P frame
  - B-frames:
    - Bidirectional frames encoded using the previous and the next I/P frame
    - Achieves maximum compression



Motion Compensation

- Exploits temporal redundancy in video frames
- Prediction:
  - Assumes that "locally" current frame can be modeled as a translation of a previous frame
  - Displacement need not be the same everywhere in the frame $\Rightarrow$ motion information must be properly encoded for reconstruction
- Bidirectional Interpolation:
  - High degree of compression
  - Areas just uncovered are not predictable from the past, but can be predicted from the future
  - Effect of noise and errors can be reduced by averaging between past and future references
  - Trade-off associated with the frequency of B-frames: increasing the frequency of B-frames decreases the correlation between the B-frame and the references, as well as between the references $\Rightarrow$ reasonable to space references by 1/10th of a second

Motion Representation

- $16 * 16$ blocks used as motion-compensation units (referred to as macro blocks)
  - Choice of macro block size is based on the tradeoff between the gain yielded by motion compensation and the cost of coding motion information
- Types of macro blocks:
  - Intra, forward-predicted, backward-predicted, average
- Two types of information are maintained:
  - Motion vector:
    - The difference between the spatial locations of the macro blocks
    - One motion vector for forward/backward predicted blocks, and two vectors for average blocks
    - Adjacent motion vectors typically differ only slightly $\Rightarrow$ encode them using differential encoding techniques (e.g., DPCM)
  - Difference between the macro block being encoded and its predictor block(s) - encode the difference using DCT-based transform coding techniques

Motion Estimation

- Block-matching techniques employed for motion estimation
- Motion vector obtained by minimizing the mismatch between block being encoded and its predictor
- Exhaustive search for such a block yield good results - but the complexity can be prohibitive
- Tradeoff between the quality of the motion vector versus the complexity of the motion estimation process is left to the implementer

Need for Hierarchical Coding/Scalable Compression

- Encode images in a manner that facilitates access to images at different quality levels or resolutions
- Progressive transmission:
  - Partial image information is transmitted in stages, and at each stage, the reconstructed image is progressively improved
  - Motivated by the need for transmitting images over low-bandwidth channels
  - Permits the progressive transmission to be stopped either if an intermediate version is of satisfactory quality or the image is found to be of no interest
  - Examples: Accessing remote multimedia databases, tele-browsing, etc.
- Multi-use environments:
  - Need for supporting a number of display devices with differing resolutions
  - Optimizes utilization of storage server and network resources
  - Example: video-in-a-window

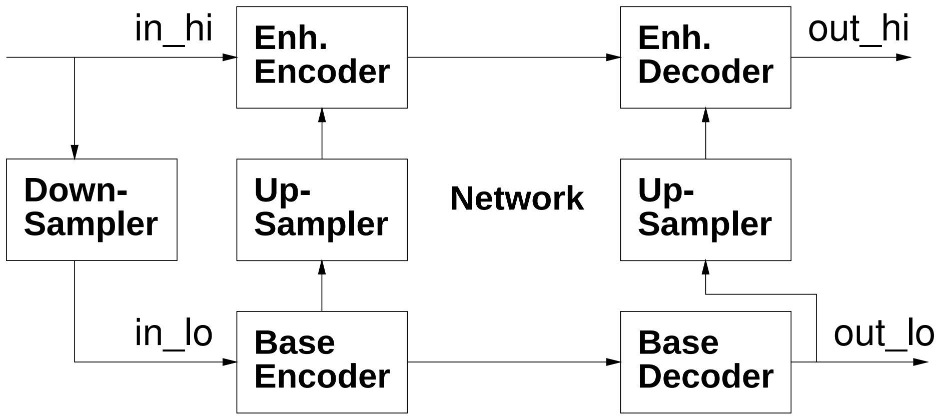
Basic Concepts

- Organize image data in the order of importance
- Fixed-resolution hierarchies:
  - Each image in the hierarchy is of the same size (i.e., identical spatial resolutions)
  - Value at a pixel is refined with each level of the hierarchy
  - Example techniques: Bit-plane encoding, transform-based hierarchical coding
- Variable-resolution hierarchies:
  - Images in the hierarchy varying their spatial resolutions $\Rightarrow$ leads to a pyramid structure
  - Base of the pyramid is the full image, each higher level results in reduction in spatial resolution of the image
  - Example techniques: sub-sampling and mean pyramids, reduced-difference pyramid

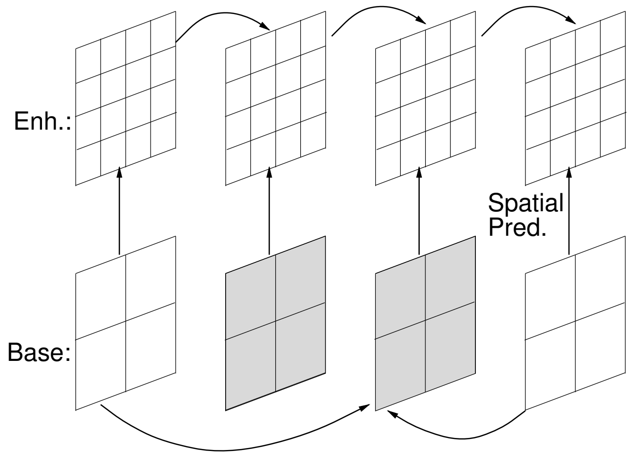
Scalable Compression in MPEG-2

- Spatial scalability
- Temporal scalability
- Chroma scalability
  - SNR scalability
  - Data partitioning

Spatial Scalability

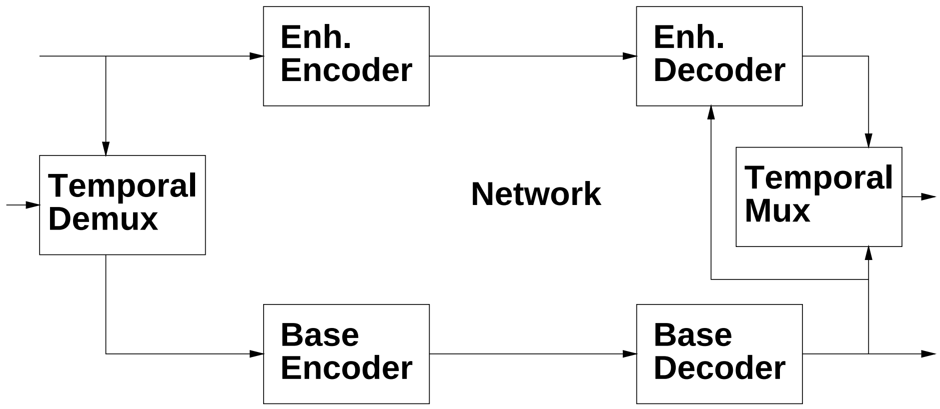


Spatial Scalability



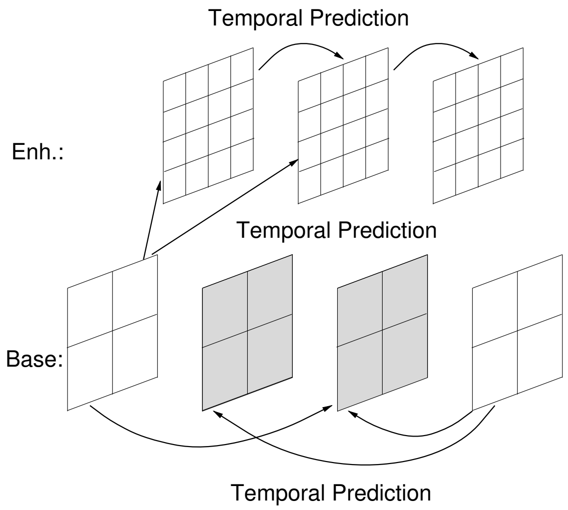
- Spatio-Temporal Weighting∗
  - predictor = w ∗ base + (1 − w) ∗ enh

Temporal Scalability

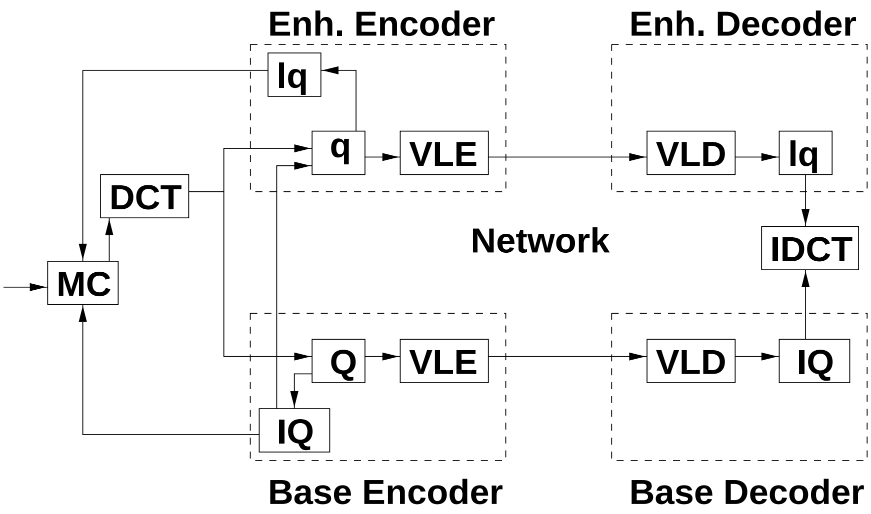


Temporal Scalability

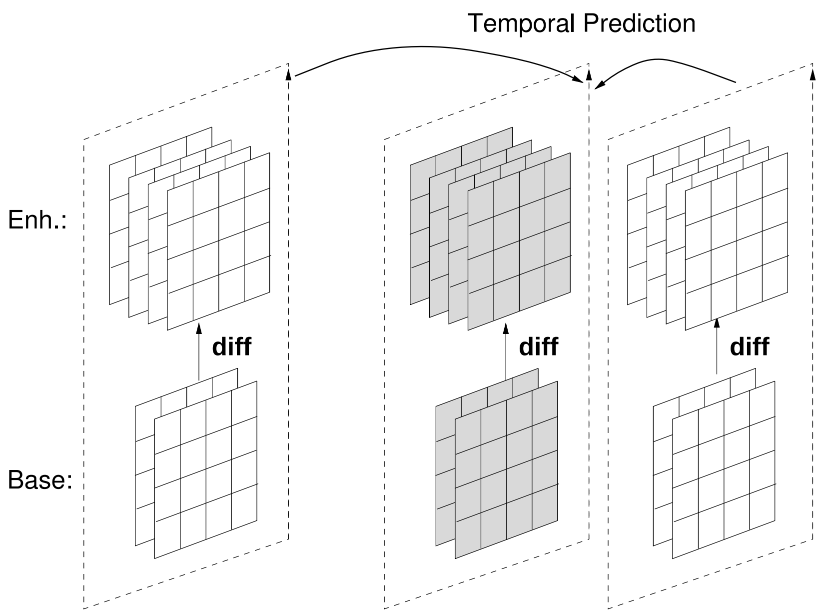
- Enhancement frames
  - Prediction using previous enhancement frame or base frame
  - Only forward prediction permitted
- Base B frames
  - Normal bidirectional prediction in base layer



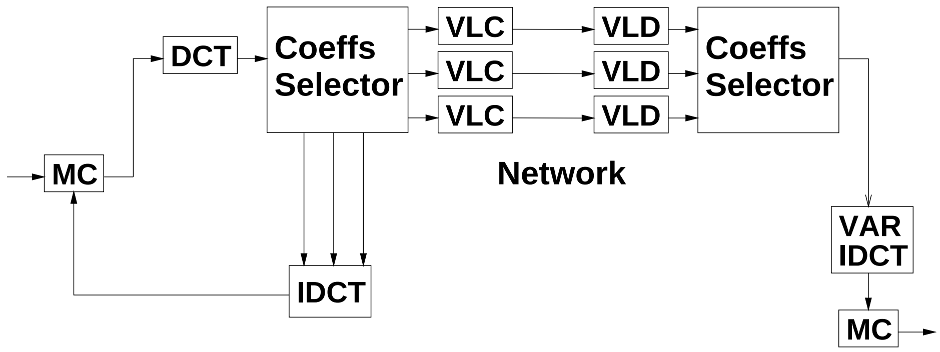
SNR Scalability



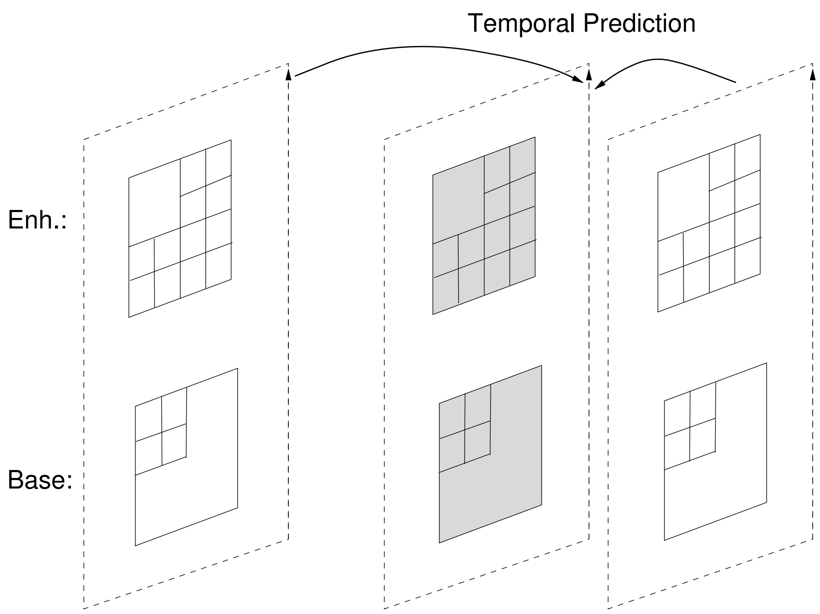
SNR Scalability



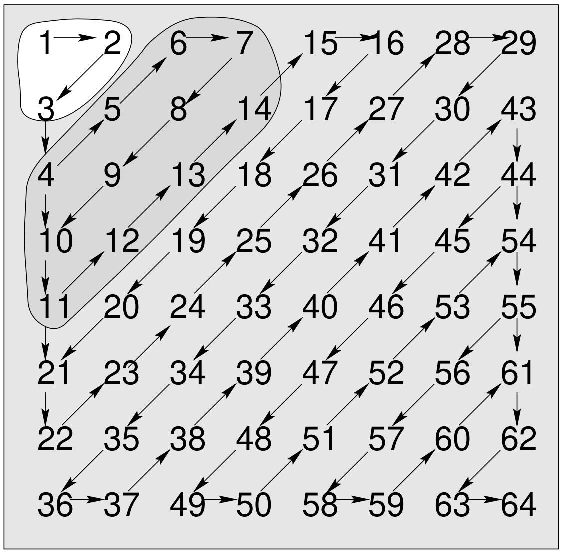
Data Partitioning/Frequency Scaling



Data Partitioning



Data Partitioning



Scalable Compression: Tradeoffs

- Advantages:
  - Facilitates multiple QoS levels
  - Resilience to packet loss
  - Backward compatible
  - Eliminates redundancy of simulcast
- Disadvantages:
  - Increased compression and decompression complexity
  - Decreased compression efficiency

Choosing a Compression Technique

- Dependent on the application requirements
- Criteria:
  - Sensitivity to input image types: global vs. local Huffman encoding, ...
  - Operational bit rate: cost-performance tradeoffs, need for operating at variable bit rates ...
  - Constant bit rate vs. constant quality: resource management issues ...
  - Implementation issues: level of compression, complexity, memory requirements, ...
  - Encoded/decoder asymmetry: interactive vs. stored information retrieval applications ...
  - Channel error tolerance: resilience to loss, cost vs. performance tradeoff ...
  - Artifacts of compression: random noise vs. jaggedness of edges
  - Progressive transmission capability: support for hierarchical coding ...