Image Compression Standards I

Prof. Dr. Touradj Ebrahimi

JPEG (*Joint Photographic Experts Group*)

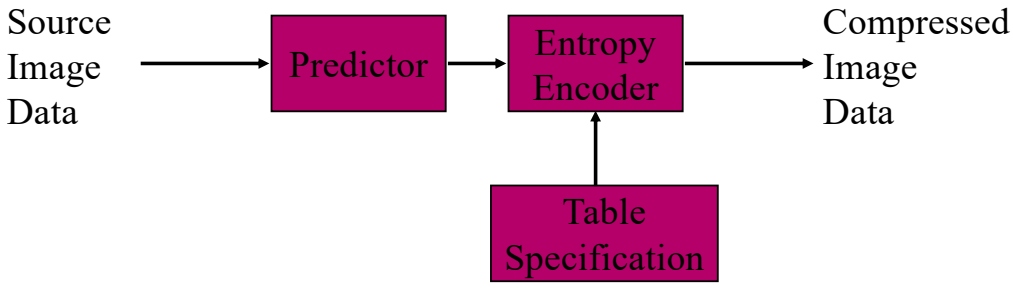
"Digital Compression and Coding of Continuous-tone Still Images"

- Joint effort between ISO et ITU-T
- Published in seven parts:
  - Part I : *Requirements and guidelines*
  - Part II: *Compliance testing*
  - Part III: *Extensions*
  - Part IV: *Registration Authorities*
  - Part V: *File Interchange Format*
  - Part VI: *Application to printing systems*
  - Part VII: *Reference Software*

Highlights

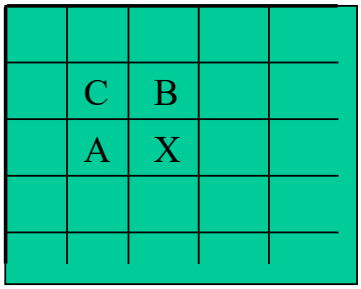
- Predictive coding based on DPCM
- No quantization step
- Huffman coding of prediction error
- Linear prediction from 3 causal neighboring pixels
- 8 different prediction modes are possible for an entire image
- Modest compression ratios of around 2:1
- Very sensitive to errors

Block diagram



Prediction modes in JPEG lossless

Mode	Prediction
0	No prediction
1	A
2	B
3	C
4	A+B-C
5	A+(B-C)/2
6	B+(A-C)/2
7	(A+B)/2



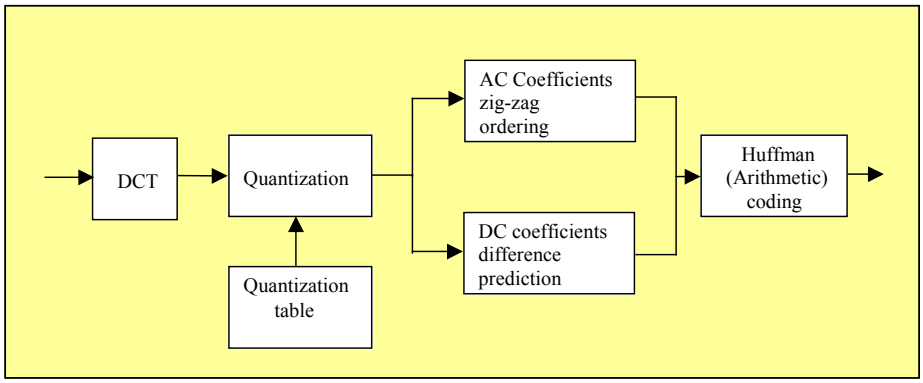
Highlights

- ⇨Transform RGB into YUV color space
- ⇨U and V components are sub-sampled prior to coding
- ⇨Divide the image into blocks of 8X8 pixels for every Y, U and V
- ⇨Every block is transformed by a 2-D 8X8 DCT
- ⇨Weighted quantization is applied to each transformed block
- ⇨A scaling factor called Q-factor is applied to quantization in order to tune the desired quality of compressed image

Highlights

- ⇨ Quantized DC coefficient of each block is coded using DPCM by taking its previous corresponding block's quantized DC value as predictor
- ⇨ Quantized AC coefficients in every block are zig-zag scanned
- ⇨ Zig-zagged coefficients are coded by a 2D Huffman coding combining amplitude of non-zero quantized coefficients and number of zero runs
  - ⇨ Arithmetic coding can be used as an option
- ⇨ Last zero runs is indicated by a special EOB code
- ⇨ Compression ratios of around 10-20:1

Block diagram



JPEG - progressive

- There are two possible scan orders:

Spectral selection
1. DC coefficient
2. Low frequency AC coefficients
3. High frequency AC coefficients

Bit-plane encoding
1. DC coefficient
2. MSBs of AC coefficients
3. LSBs of AC coefficients

JPEG - progressive

- Progressive versus sequential transmission

Industry standards based on JPEG

- JFIF (*JPEG File Interchange Format,* `<xxxxxx.jpg>`)
- JTIP (*JPEG Tiled, Pyramid Format*)
- TIFF (*Tagged Image File Format*)
- SPIFF (*Still Picture Interchange File Format, JPEG Part 3*)
- FlashPix
  - Developed by Hewlett-Packard, Kodak, Microsoft, Live Picture (1996)
  - Transferred to Digital Imaging Group (DIG)

JPEG 2000

JPEG 2000 parts

- Part I: A set of tools covering a good proportion of application requirements (JP2)
- Other parts are also defined and planned for a further date
- Amendment are added to Part I with additional profiles
- Schedule for part I:

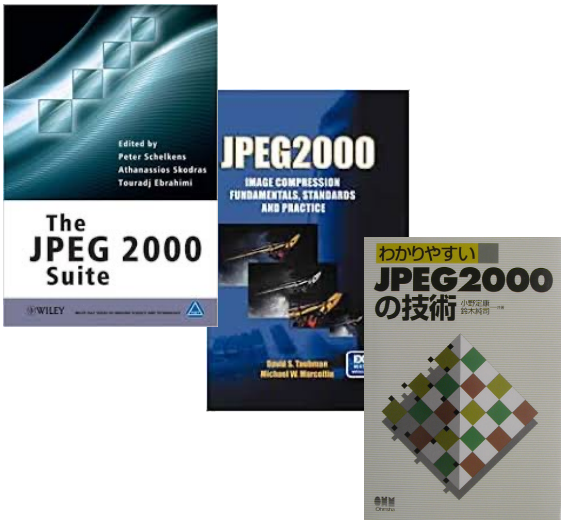
Elevation to FDIS: *08/00*
Elevation to IS: *12/00*

JPEG 2000 parts (cont'd)

- Part II: Extension tools to cover specific applications (JPX)
- Part III: Motion JPEG 2000 (MJP)
- Part IV: Conformance
- Part V: Reference software
- Part VI: Compound images file format (JPM)
- Part VII: EMPTY
- Part VIII: Secure JPEG 2000 (JPSEC)
- Part IX: Interactivity and Protocols (JPIP)
- Part X: Volumetric (JP3D)
- Part XI: Wireless (JPWL)
- Part XII: ISO media file format
- Part XIII: Reference encoder specification
- Part XIV: XML syntax description
- Part XV: High throughput JPEG 2000 (HTJ2K)
- Part XVI: Wrapping JPEG 2000 in HEIF
- Part XVII: Coding of discontinuous media

JPEG2000 Features in Part I

- High compression efficiency
- Lossless colour transformations
- Lossy and lossless coding in one algorithm
- Embedded lossy to lossless coding
- Progressive by resolution, quality, position, …
- Static and dynamic Region-of-Interest coding/decoding
- Error resilience
- Perceptual quality coding
- Multiple component image coding
- Tiling
- Palletized image coding
- Light file format (optional)
- …



JPEG at 0.125 bpp (192:1)

JPEG 2000 at 0.125 bpp (192:1)

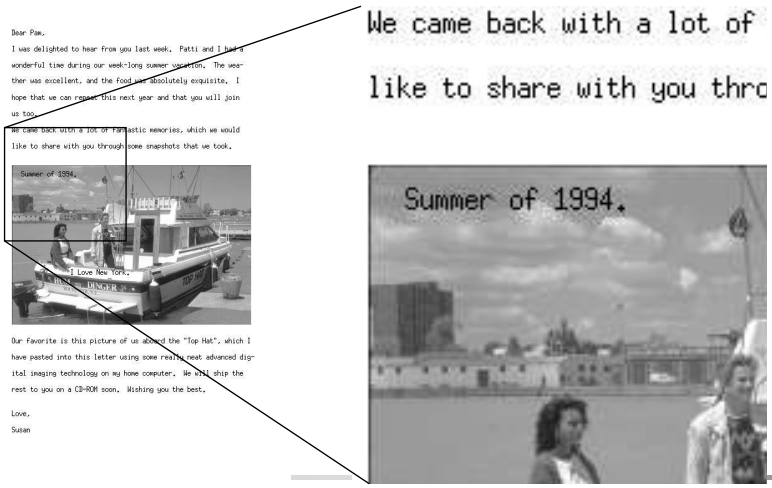
JPEG at 0.25 bpp (96:1)

JPEG 2000 at 0.25 bpp (96:1)

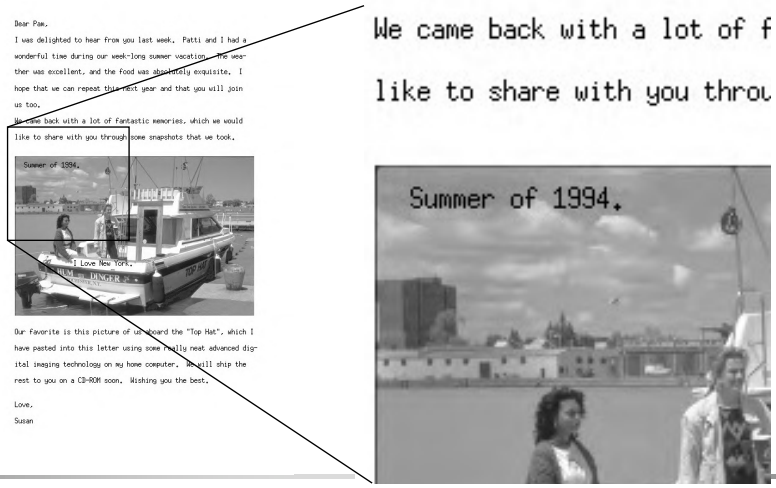
JPEG at 0.5 bpp (48:1)

JPEG 2000 at 0.5 bpp (48:1)

JPEG at 1 bpp (8:1)

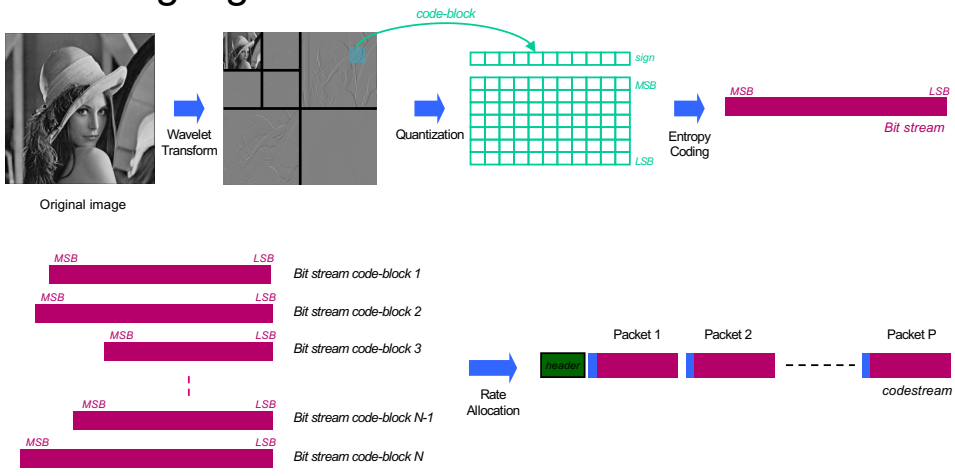


JPEG 2000 at 1 bpp (8:1)



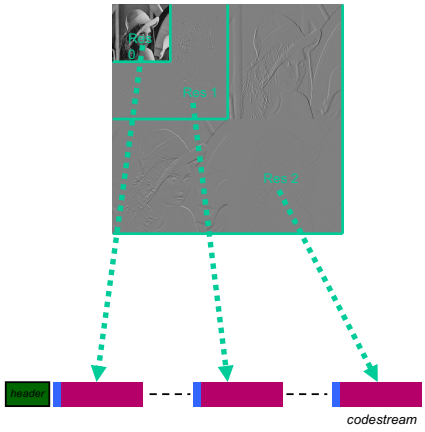
JPEG 2000 overview

- Encoding algorithm:

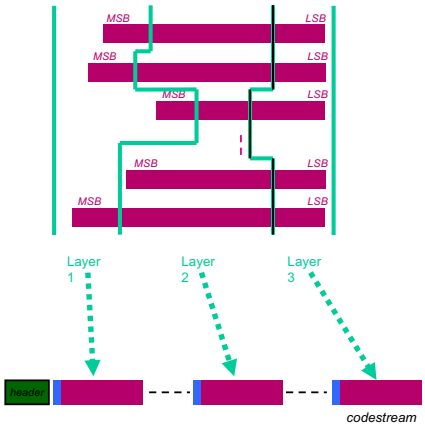


JPEG 2000 progressive coding

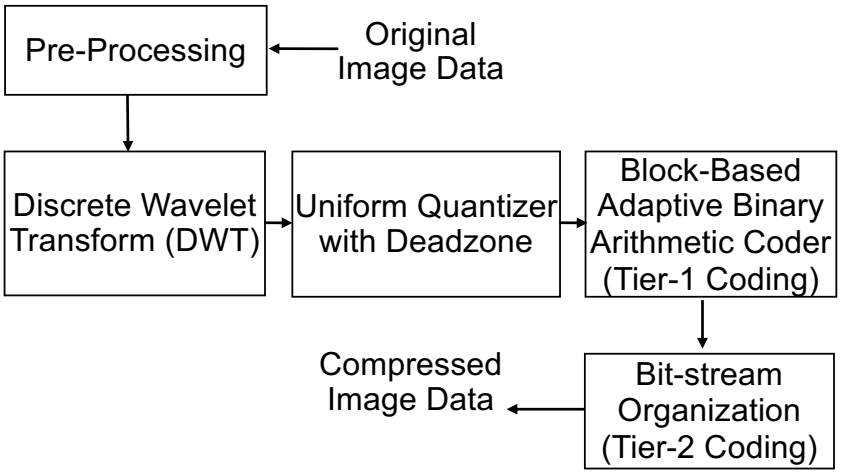
Progressive by resolution



Progressive by quality



JPEG 2000 fundamental building blocks



Irreversible Color Transform (ICT)

The ICT is the same as the conventional YCbCr transform for the representation of image and video signals:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.4021 \\ 1.0 & -0.3441 & -0.7142 \\ 1.0 & 1.7718 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix}$$

Reversible Color Transform (RCT)

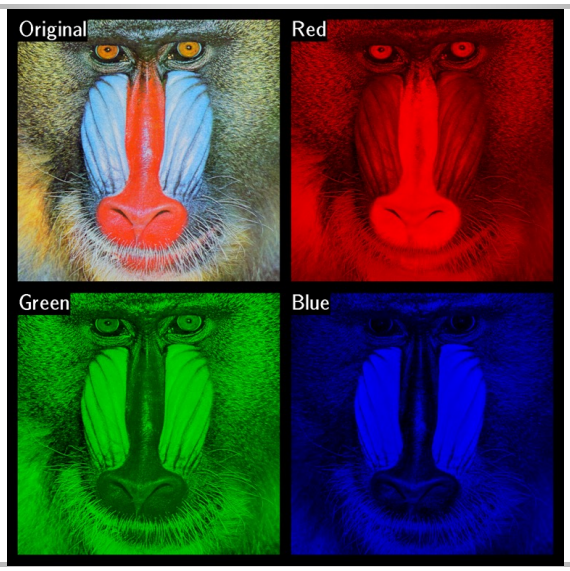
- The ICT is not capable of lossless coding
  - The reversible color transform (RCT) is an integer-to-integer approximation intended for lossless coding.

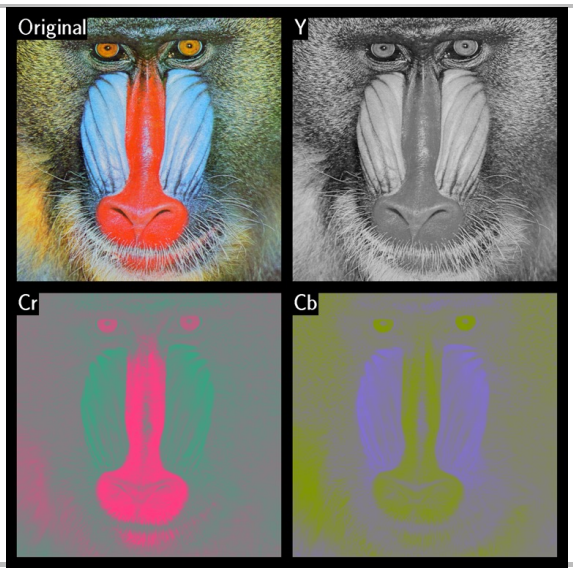
- Forward RCT:

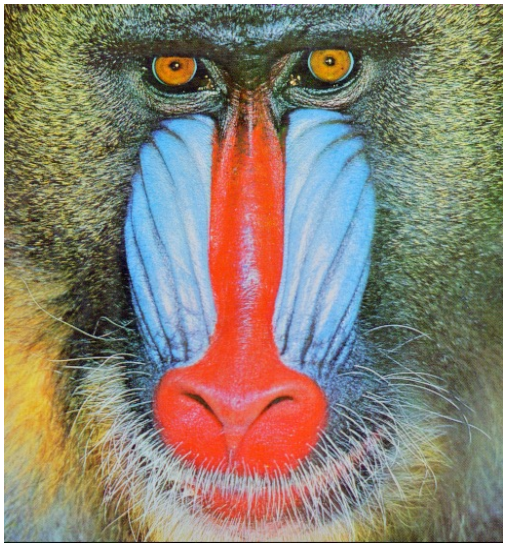
$$Y = \left\lfloor \frac{1}{4}(R + 2G + B) \right\rfloor$$
$$C_b = B - G$$
$$C_r = R - G$$

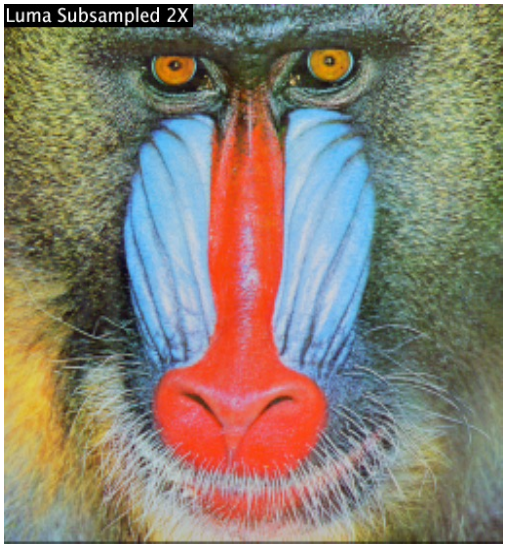
- Inverse RCT:

$$G = Y - \left\lfloor \frac{1}{4}(C_b + C_r) \right\rfloor$$
$$R = C_r + G$$
$$B = C_b + G$$


Original
Red
Green
Blue

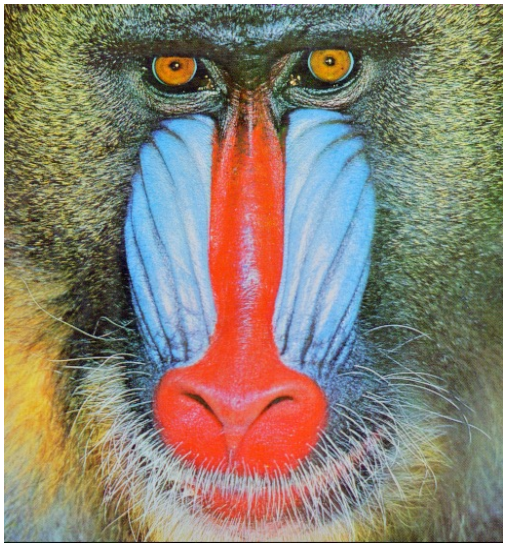

Original
Y
Cr
Cb



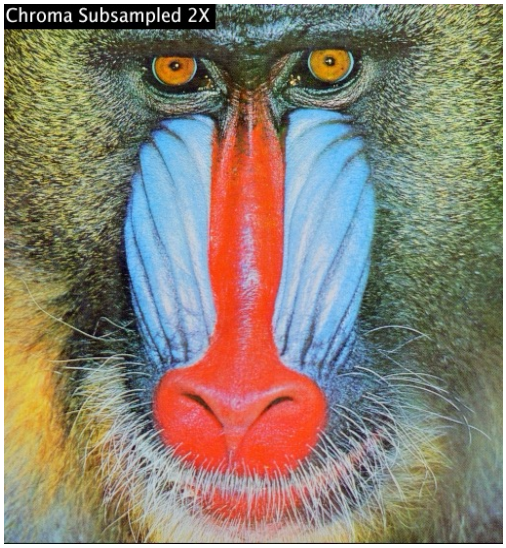

Luma Subsampled 2X


Luma Subsampled 4X

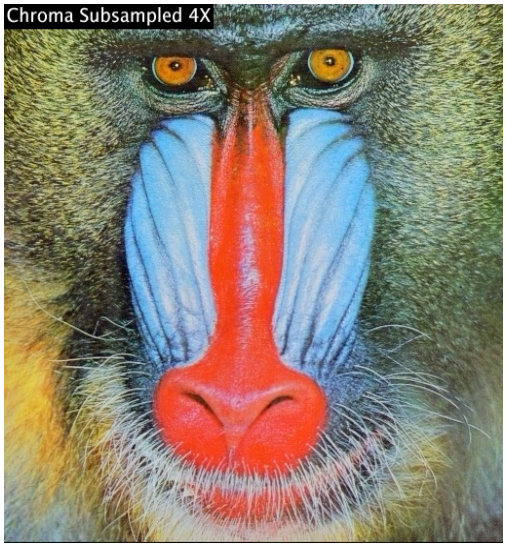

Luma Subsampled 8X

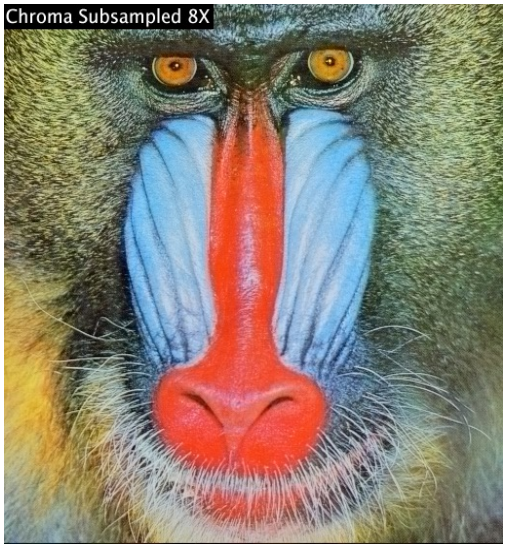


35

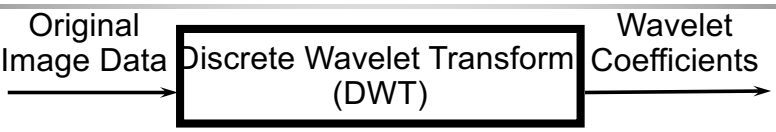


36


Chroma Subsampled 4X


Chroma Subsampled 8X

Transformation in JPEG 2000 Part I



- Multi-resolution image representation is inherent to DWT.
- The full-frame nature of the transform decorrelates the image across a larger scale and eliminates blocking artifacts at high compression.
- Use of integer DWT filters allows for both lossless and lossy compression within a single compressed bit-stream.
- DWT provides a frequency band decomposition of the image where each subband can be quantized according to its visual importance.
- Two filters in Part I: irreversible Daubechies (9,7) and reversible (5,3)
- Part II allows arbitrary filters

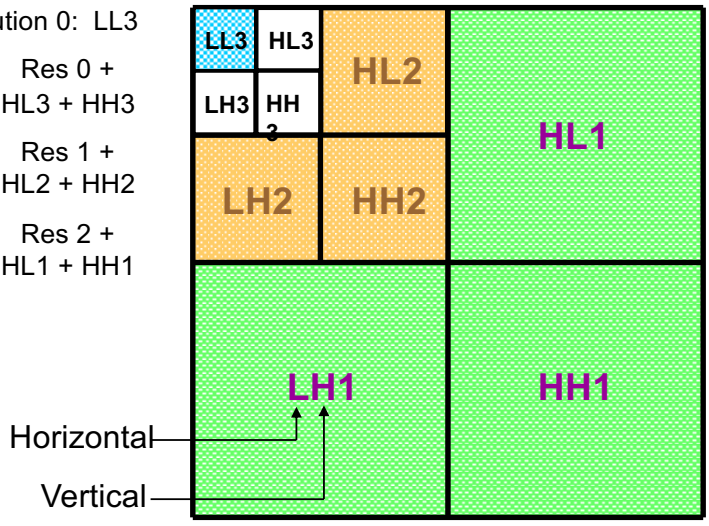
2-D wavelet decomposition

Resolution 0: LL3

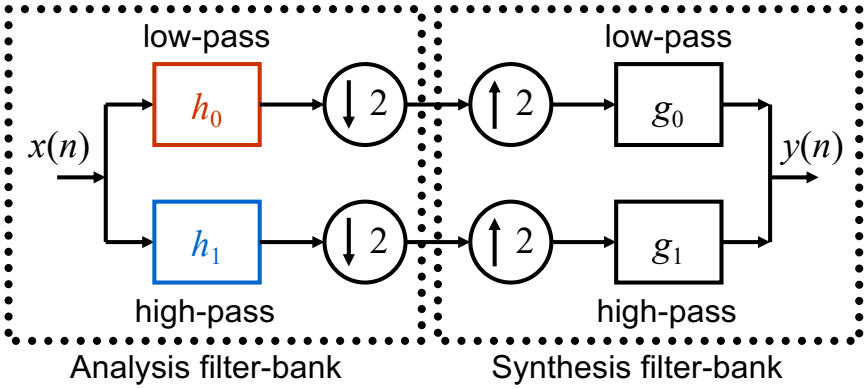
Res 1: Res 0 + LH3 + HL3 + HH3

Res 2: Res 1 + LH2 + HL2 + HH2

Res 3: Res 2 + LH1 + HL1 + HH1



Bi-orthogonal DWT



Bi-orthogonal filter bank:

h_0 is orthogonal to g_1

h_1 is orthogonal to g_0

The JPEG 2000 DWT filters

- Irreversible Daubechies (9,7)

n	$h_0(n)$	n	$h_1(n)$
0	+0.602949018236	−1	+1.115087052456
±1	+0.266864118442	−2, 0	−0.591271763114
±2	−0.078223266528	−3, 1	−0.057543526228
±3	−0.016864118442	−4, 2	+0.091271763114
±4	+0.026748757410		

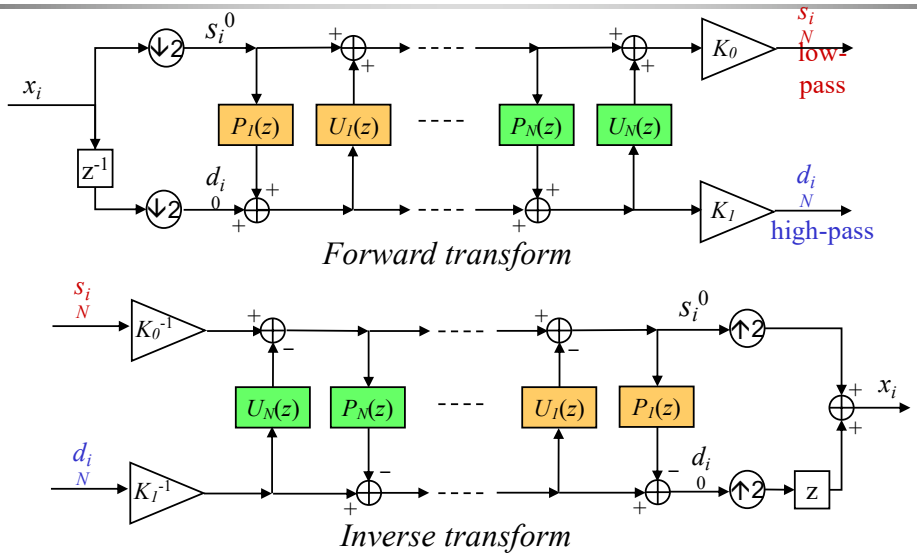
- Reversible (5,3), derived from Le Gall (5,3)

Le Gall (5,3)
(not exactly JPEG 2000's)

n	$h_0(n)$	n	$h_1(n)$
0	+6/8	−1	+1
±1	+2/8	−2, 0	−1/2
±2	−1/8		

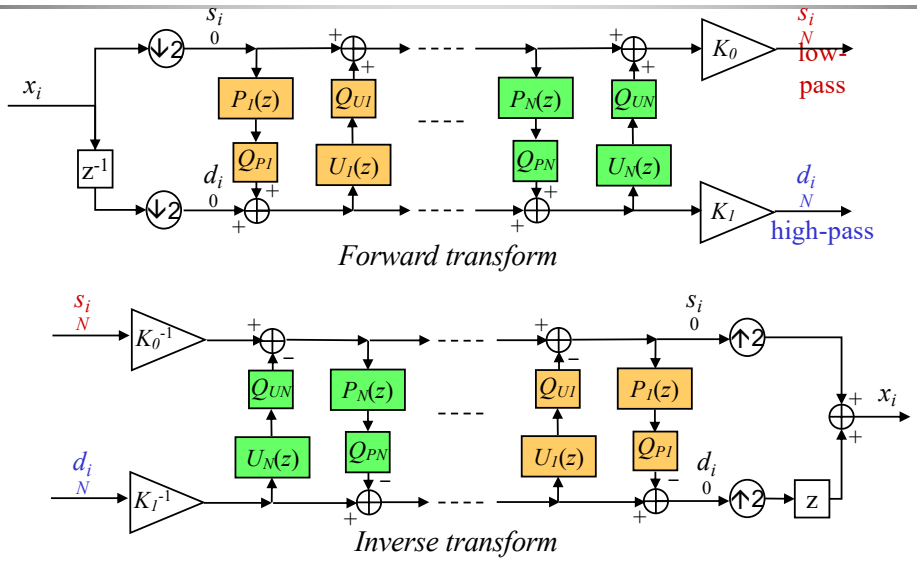
- In addition, Part II allows for arbitrary filters

Lifting block diagram



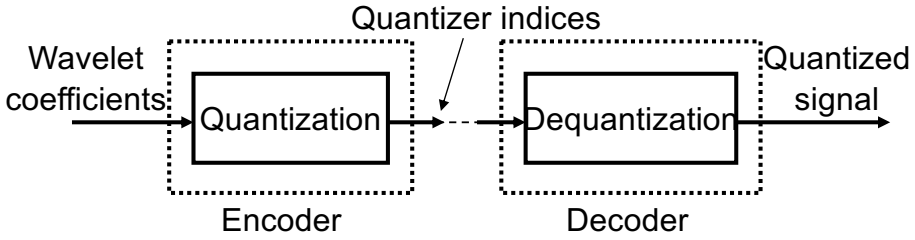
43

Integer-to-integer transforms



44

Quantization in JPEG2000 Part I



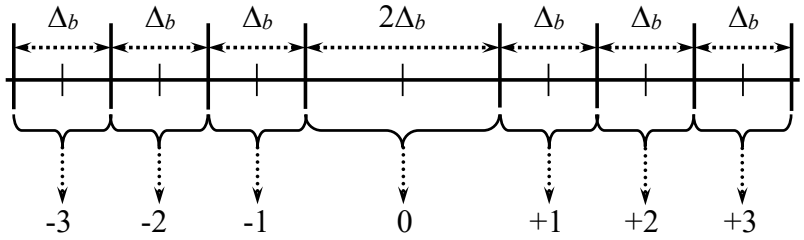
- Uniform quantization with deadzone is used to quantize all the wavelet coefficients.
- For each subband *b*, a basic quantizer step size D_b is selected by the user and is used to quantize all the coefficients in that subband.
- The choice of the quantizer step size for each subband can be based on visual models, such as the contrast sensitivity function (CSF). This gives higher compression ratio for same visual quality.

Uniform scalar quantizer with deadzone

- Quantization rule:

$$q = sign(y)\left\lfloor \frac{|y|}{\Delta_b} \right\rfloor$$

where *y* is the input to the quantizer, Δ_b is the quantizer step size, *q* is the resulting quantizer index, sign(*y*) denotes the sign of *y*, $|y|$ denotes the absolute value of *y*, and $\lfloor x \rfloor$ denotes the largest integer not larger than *x*.



Dequantization rule

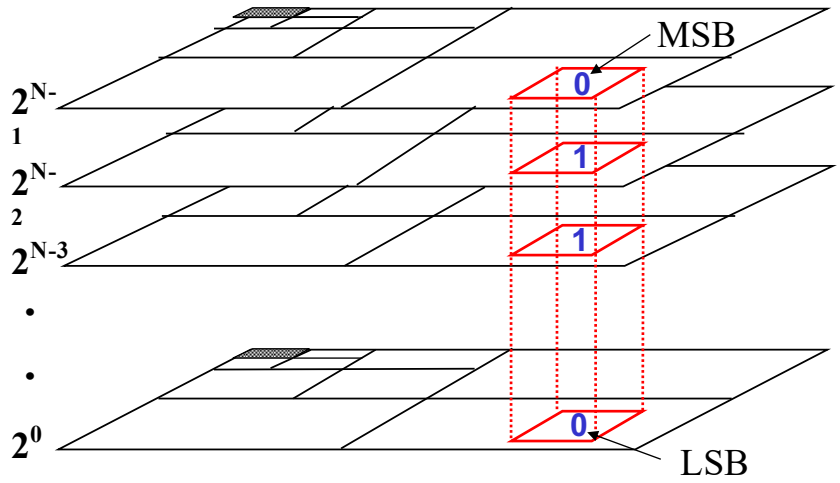
- Dequantization rule:

$$z = \begin{cases} [q + r\,sign(q)]\Delta_b & \text{for } q \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

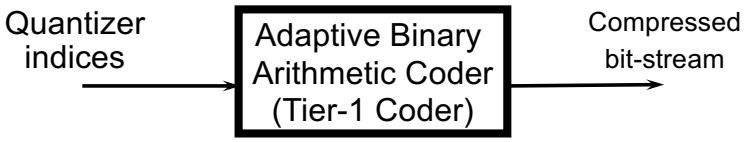
where q is the quantizer index, Δ_b is the quantizer step size, z is the reconstructed (quantized) signal value, sign(q) denotes the sign of q, and r is the reconstruction bias.

- $r = 0.5$ results in midpoint reconstruction (no bias).
- $r < 0.5$ biases the reconstruction towards zero. A popular value for r is 0.375.
- In JPEG 2000 Part I, the parameter r is arbitrarily chosen by the decoder.

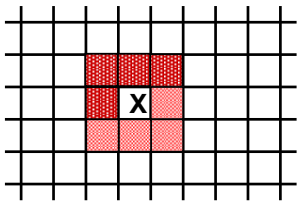
Embedded quantization by bit-plane coding



Entropy coding in JPEG 2000 Part I



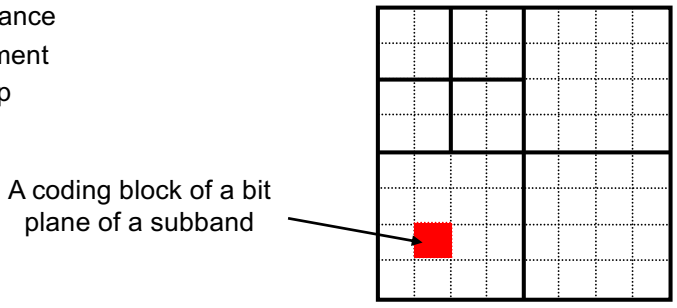
- Context-based adaptive binary arithmetic coding is used in JPEG 2000 to efficiently compress each individual bit plane.



- The binary value of a sample in a block of a bit plane of a subband is coded as a binary symbol with the JBIG-2 **MQ-Coder**.

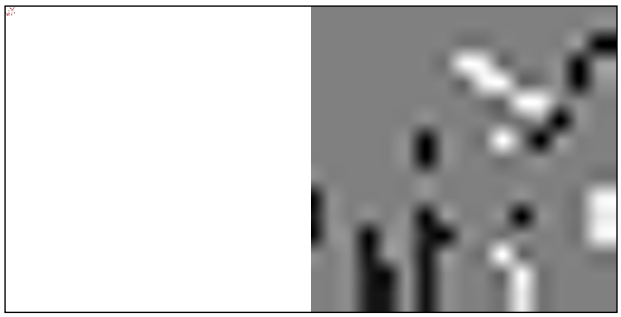
JPEG 2000 entropy coder

- Each bit plane is further broken down into blocks (e.g., 64 × 64). The blocks are coded independently (i.e., the bit-stream for each block can be decoded independent of other data) using three coding passes. The coding progresses from the most significant bit-plane to the least significant bit-plane.
  - Significance
  - Refinement
  - Cleanup



Significance =	0
Refine =	0
Cleanup =	21
Total Bytes	21

Bit plane **1**
Compression ratio = 12483 : 1
RMSE = 39.69 PSNR = 16.16 db
% refined = 0 % insig. = 99.99



red: cleanup; green: significance; black: refinement; white: non-significant

Significance =	38
Refine =	13
Cleanup =	57
Total Bytes	108

Bit plane **3**
Compression ratio = 1533 : 1
RMSE = 21.59 PSNR = 21.45 db
% refined = 0.05 % insig. = 99.89

red: cleanup; green: significance; black: refinement; white: non-significant

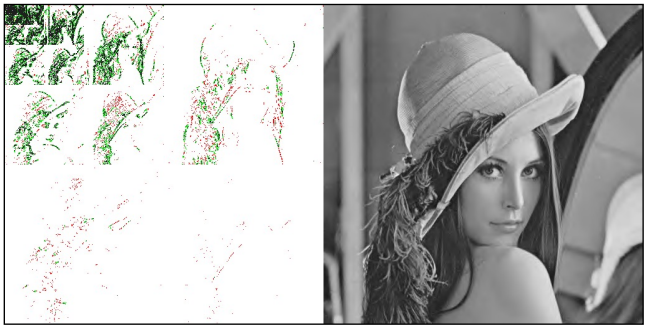
Significance =	224
Refine =	73
Cleanup =	383
Total Bytes	680

Bit plane **5**
Compression ratio = 233 : 1
RMSE = 12.11 PSNR = 26.47 db
% refined = 0.23 % insig. = 99.43

red: cleanup; green: significance; black: refinement; white: non-significant

Significance =	2315
Refine =	932
Cleanup =	2570
Total Bytes	5817

Bit plane **8**
Compression ratio = 23 : 1
RMSE = 4.18 PSNR = 35.70 db
% refined = 2.91 % insig. = 93.99



red: cleanup; green: significance.; black: refinement; white: non-significant

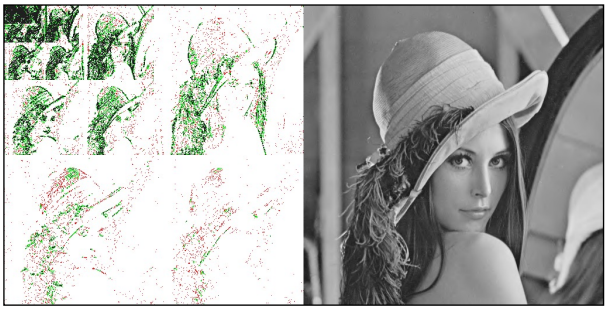
Significance =	4593
Refine =	1925
Cleanup =	5465
Total Bytes	11983

Bit plane **9**

Compression ratio = 11.2 : 1

RMSE = 2.90 PSNR = 38.87 db

% refined = 6.01 % insig. = 87.66

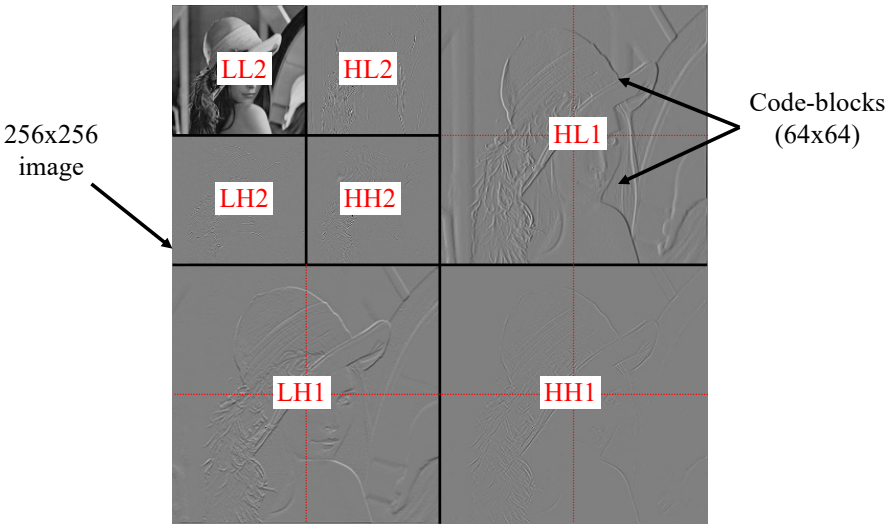


red: cleanup; green: significance; black: refinement; white: non-significant

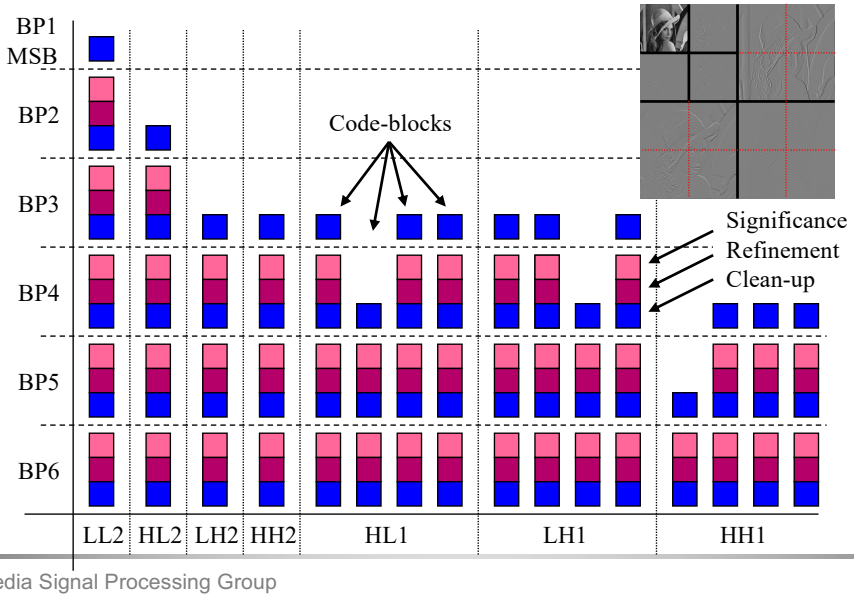
Tier 2 role

- **Tier 1** generates a collection of bitstreams
  - One independent bitstream for each code-block
  - Each bitstream is embedded
- **Tier 2** multiplexes the bitstreams for inclusion in the codestream and signals the ordering of the resulting coded bitplane passes in an efficient manner
- Tier 2 coded data can be rather easily parsed
- Tier 2 enables SNR, resolution, spatial, ROI and arbitrary progression and scalability

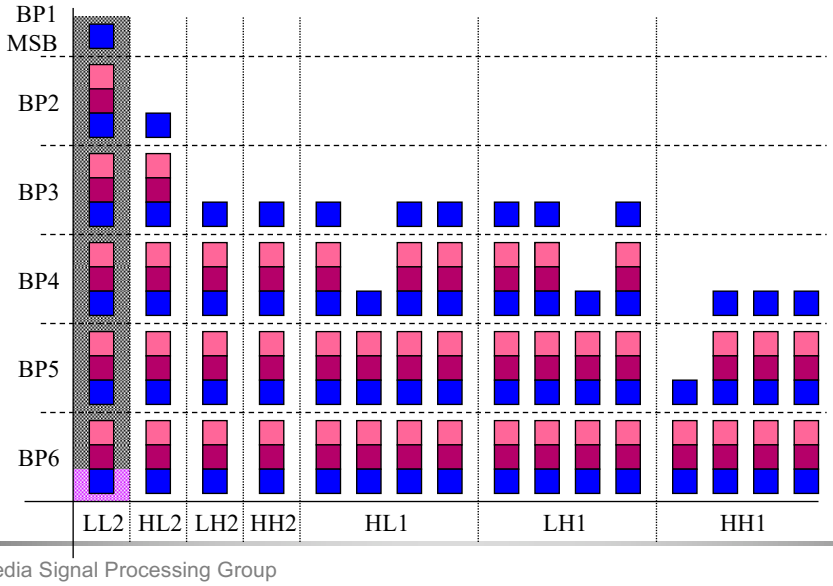
Example of bit-plane pass coded data



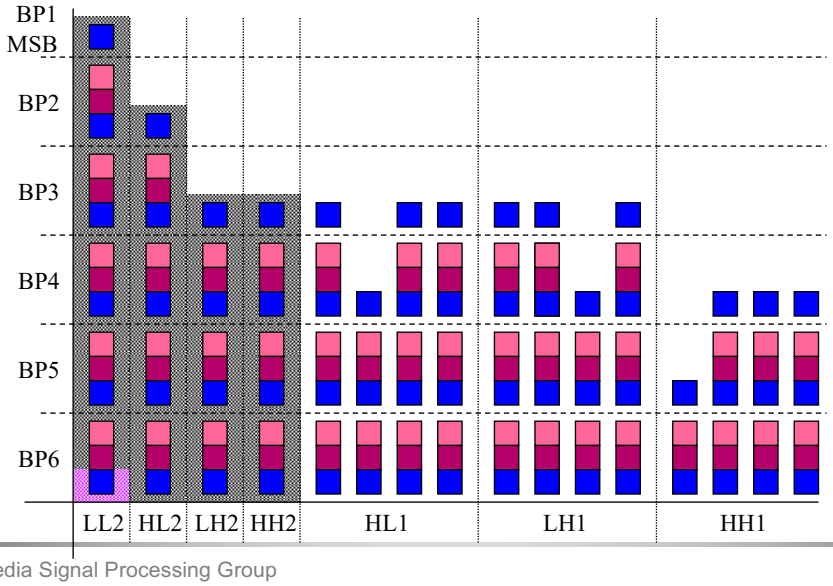
Example of bit-plane pass coded data



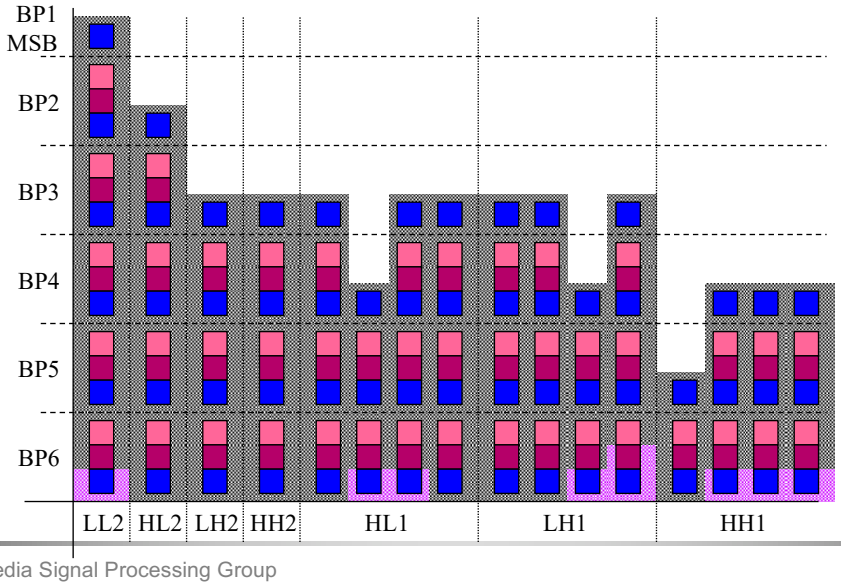
Lowest resolution, Highest quality



Medium resolution, Highest quality

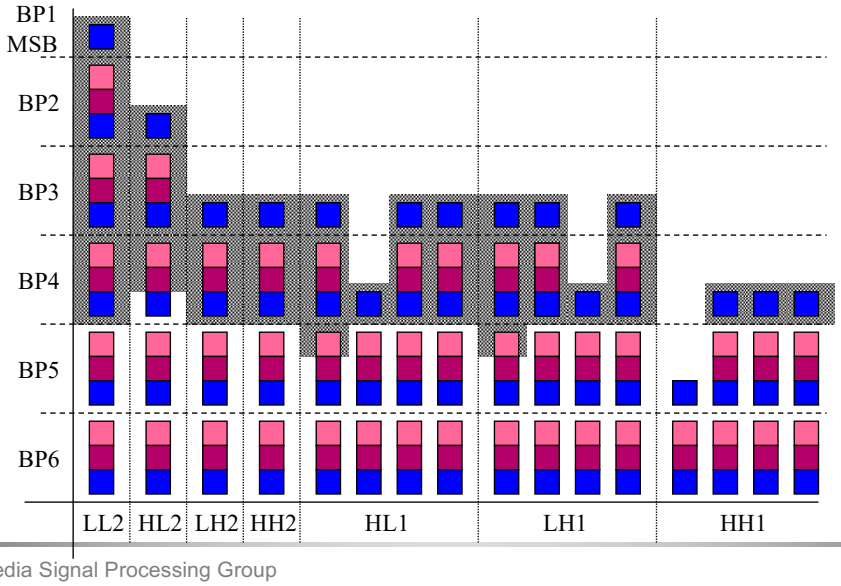


Highest resolution, Highest quality



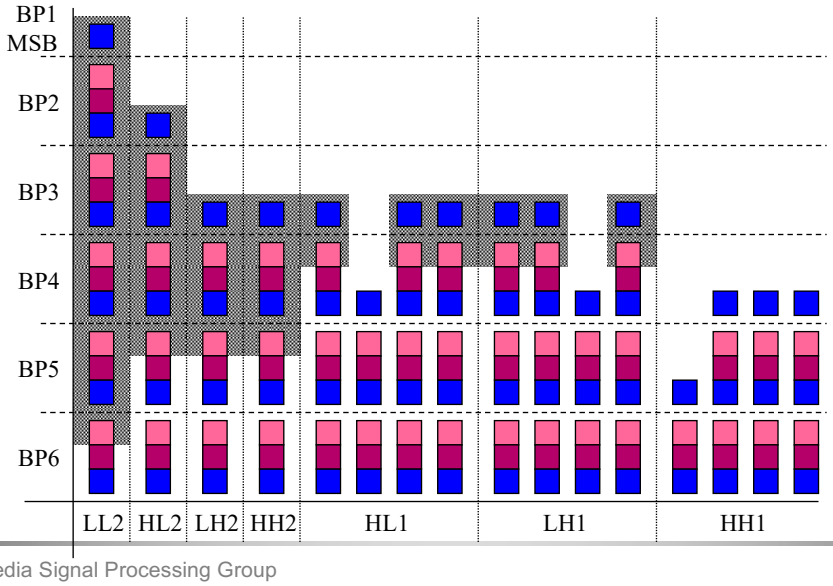
61

Highest resolution, Target SNR quality



62

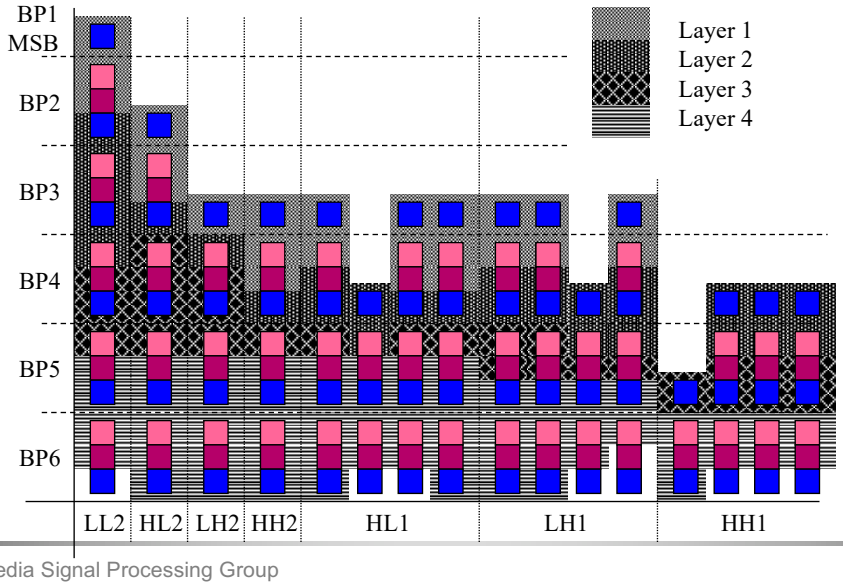
Highest resolution, Target Visual quality



Layers

- **Layer**: a collection of some consecutive bit-plane coding passes from all code-blocks in all subbands and components. Each code-block can contribute an arbitrary number of bit-plane coding passes to a layer.
- Each layer successively increases the image quality. Most often associated with SNR or visual quality levels.
- Layers are explicitly signalled and can be arbitrarily determined by the encoder
- The number of layers can range from 1 to 65535. Typically around 20. Larger numbers are intended for interactive sessions were each layer is generated depending on user feedback.

Example of layer organization

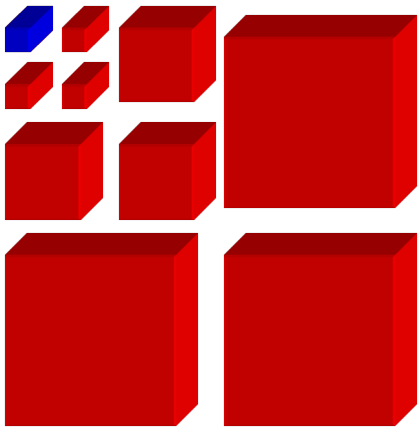


Layer (SNR) progressive example

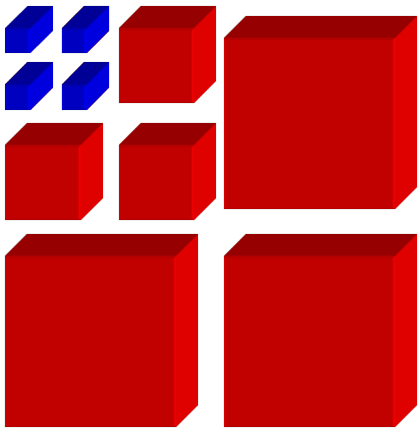
Layer (SNR) progressive example

Layer (SNR) progressive example

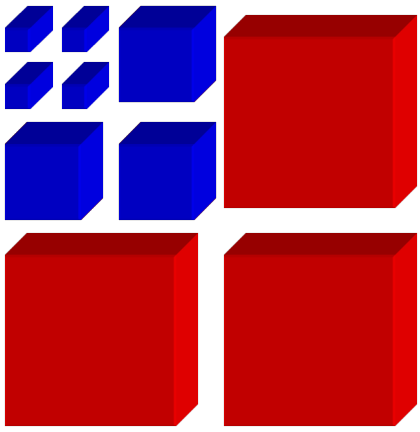
Resolution progressive example



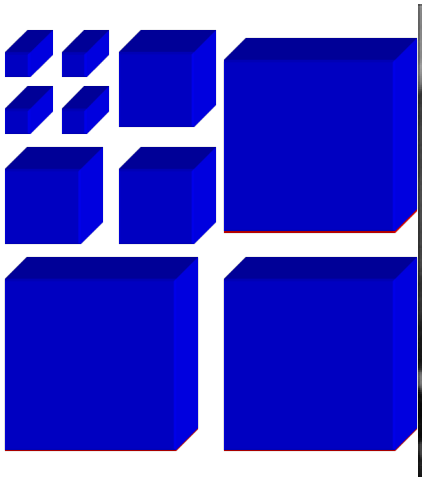
Resolution progressive example



Resolution progressive example



Resolution progressive example



Packets

- **Packet**: compressed data representing a specific tile, layer, component and resolution level.
- Packet header signals
  - Which code-blocks are included in the packet
  - The number of most significant all zero bit-planes skipped by the entropy coder, for each newly included code-block
  - The number of included coding-passes for each code-block
  - The length of included coded data for each code-block
- Packet body: concatenation of included coded image data

Codestream

- **Codestream**: compressed image data with all the signaling required to properly decompress it.
- Composed of a main and tile headers, that specify coding parameters in a hierarchical way, plus the encoded data for each tile.
- The compressed data for a tile can be broken up in tile-parts, and the different tile-parts interleaved in the codestream to allow for non-tile progressiveness.
- The codestream is the minimum exchange format for JPEG 2000 encoded data, but usually the codestream is embedded in a file format.

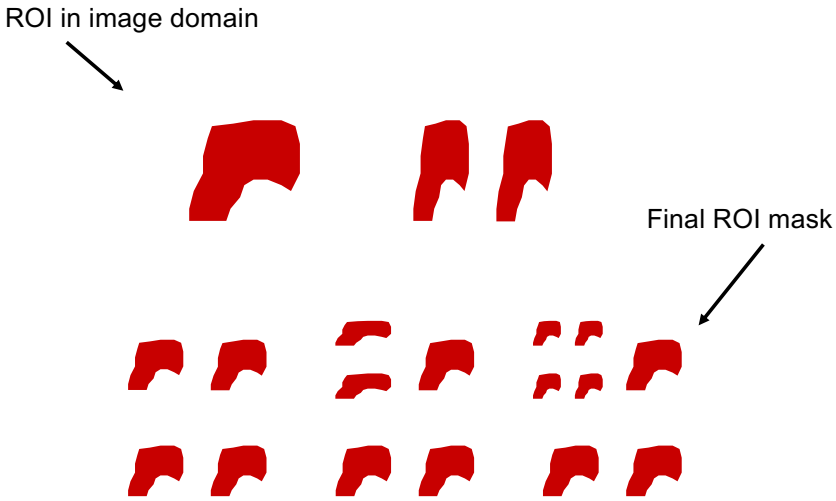
Rate allocation

- Rate allocation is the process that allows to target a specific compression ratio with the best possible quality (MSE, visual or other) for each layer and/or entire codestream. Possible types are:
  - None: compression ratio is determined solely by the quantization step sizes and image content.
  - Iterative: quantization step sizes are adjusted according to obtained compression ratio and operation is repeated.
  - Post-compression: rate allocation is performed after the image data has been coded, in one step.
  - Others (Lagrangian, scan-based, etc.)
- Not standardized by JPEG 2000 $\Rightarrow$ encoder choice.

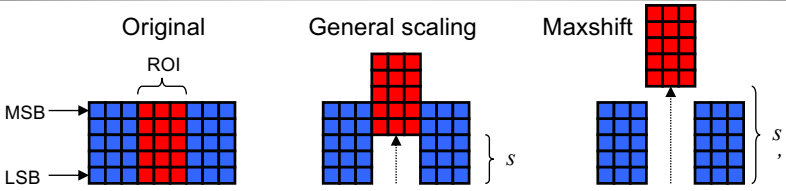
Region of Interest coding principle

- **Region of Interest** (ROI) coding allows a non-uniform distribution of quality. The ROI is coded with a higher quality than the background (BG). A higher compression ratio can be achieved with same or higher quality inside ROIs.
- **Static** ROIs are defined at encoding time and are suitable for storage, fixed transmission, remote sensing, etc. Commonly referred to as ROI coding.
- **Dynamic** ROIs are defined interactively by a user in a client/server situation during a progressive transmission. Suitable for telemedicine, PDAs, mobile communications, etc. They can be achieved by the dynamic generation of layers matching the user's request.

Lossless ROI mask generation

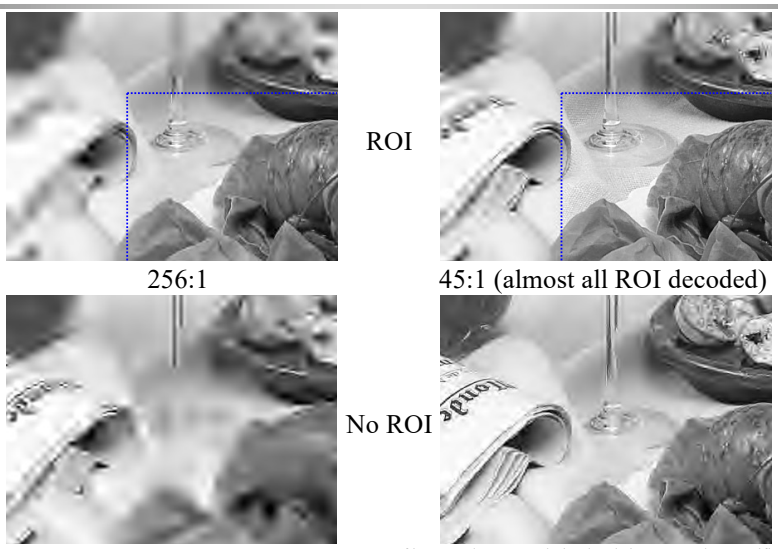


Encoding ROIs: scaling



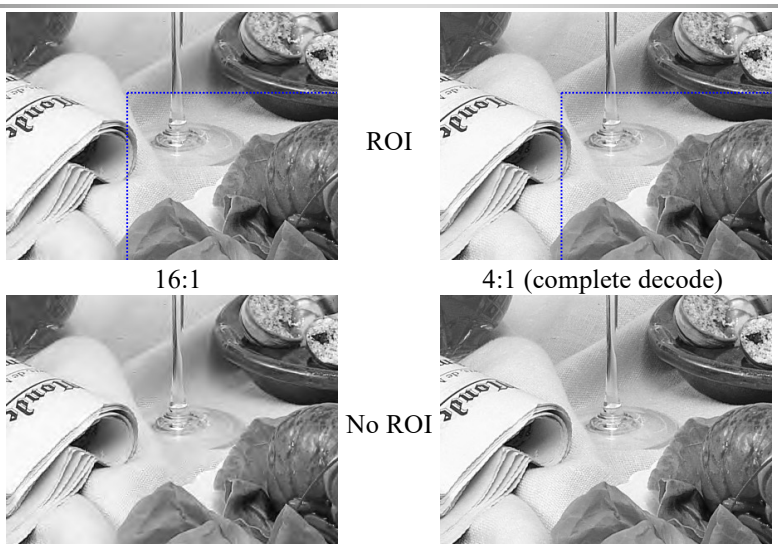
- Shift up quantized ROI coefficients by *s* bitplanes. The value of *s* is recorded in the codestream header for each ROI.
- At decoder ROI coefficients are unshifted prior to dequantization.
- The ROI mask is required at both, encoder and decoder.
- In maxshift, *s* is large enough to separate ROI and BG
  - No ROI mask required at decoder
  - ROI <-> BG quality differential not controlled

ROI Maxshift example



ROI covers 5% of image, 2 lowest resolution levels in ROI mask. Magnified portion shown.

ROI Maxshift example (cont'd)



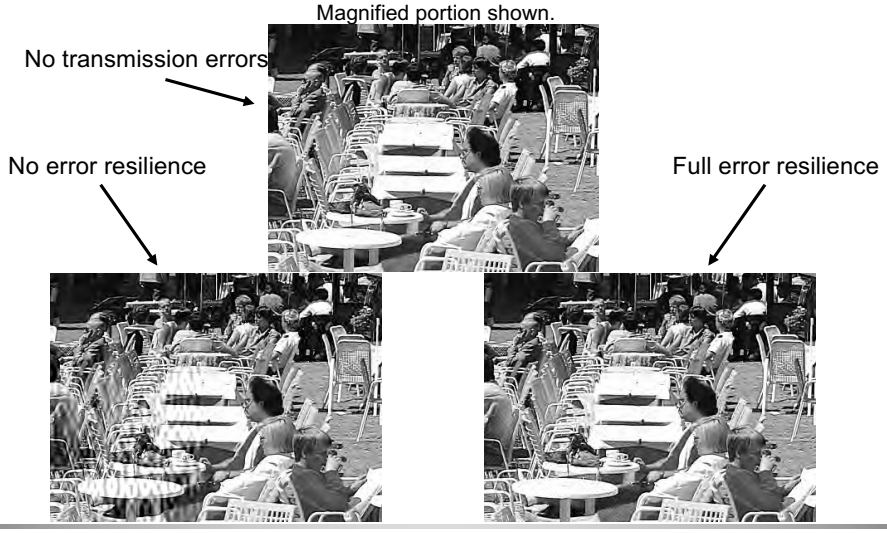
ROI covers 5% of image, 2 lowest resolution levels in ROI mask. Magnified portion shown.

Error-prone channels

- When delivering compressed images across error-prone channels any transmission error can severely affect the decoded image quality. This is specially true since variable length coding is used in the code-block entropy coding and packet heads.
- Error types can be random errors, burst error and missing bytes (i.e. network packet loss).
- Since each code-block is independently coded an error in a code-block's bitstream will be contained within that code-block. Nevertheless severe distortion can occur in the case of an error.
- Packet heads are interdependent and thus fragile.

Error resilience example

16:1 compression ratio. Transmission error rate 10^{-5}. No errors in codestream header.
Magnified portion shown.



Error protection techniques

- Coded image data
  - Code-block partition
  - Regular termination of arithmetic coder
  - Segmentation symbols
- Packet heads
  - Start of packet markers
  - Packet heads in main / tile codestream header
  - Partition of packets into precincts

JPEG 2000 File Format: JP2

- **JP2** is the optional JPEG 2000 file format to encapsulate JPEG 2000 codestreams.
  - *Extension: jp2*
  - *Allows to embed XML information (e.g., metadata)*
  - *Alpha channel (e.g., transparency)*
  - *Accurate color interpretation*
  - *"True color" and "palette color" supported*
  - *Intellectual property information*
  - *Capture and default display resolution*
  - *File "magic number"*
  - *File transfer errors (ASCII ftp, 7 bit e-mail, etc.)*

Software

- C implementation (SAIC / Univ. of Arizona / HP)
  - *Verification Model (VM) used to develop the JPEG 2000 standard*
- JJ2000 Java implementation (EPFL, Ericsson, Canon)
  - *Part V reference implementation and publicly accessible*
- JasPer C implementation (ImagePower, UBC)
  - *Part V reference implementation, and publicly accessible*
- OpenJPEG C implementation (UCL, IntoPix, …)
  - *Part V reference implementation, and publicly accessible*
- Commercial implementations
  - *Kakadu*
  - *Apple*
  - *Adobe*