Image compression

- COMPRESSION
- IMAGE COMPRESSION STANDARDS

Why compression?

To reduce the volume of data to be transmitted

- text, fax, images, videos, audio
- the bandwidth required for transmission
- storage requirements.

Frame size	720*480 pixels
Frame rate	30 frame/second
Bits per pixel	24

Example: for standard definition TV

Data rate: $30 * (720 * 480) * 24 / 8 = 31{,}104{,}000 \text{ bytes/sec}$

For 2-hour SDTV clip, the required storage:

$$31{,}104{,}000 * (60^2) * 2 \cong 2.24 * 10^{11} \text{bytes}$$

Redundancy

Three types of data redundancy:

1. Coding redundancy
   - Unnecessarily assign a long code to a symbol, e.g. fixed-length coding

2. Spatial and temporal redundancy, e.g. pixel replication
   - Due to the spatial correlation between pixels of an image, e.g. A large area of one intensity.
   - Due to the temporal correlation between pixels of a video, e.g. Between two consecutive frames.

3. Irrelevant information
   - Information that is ignored by Human Visual System.

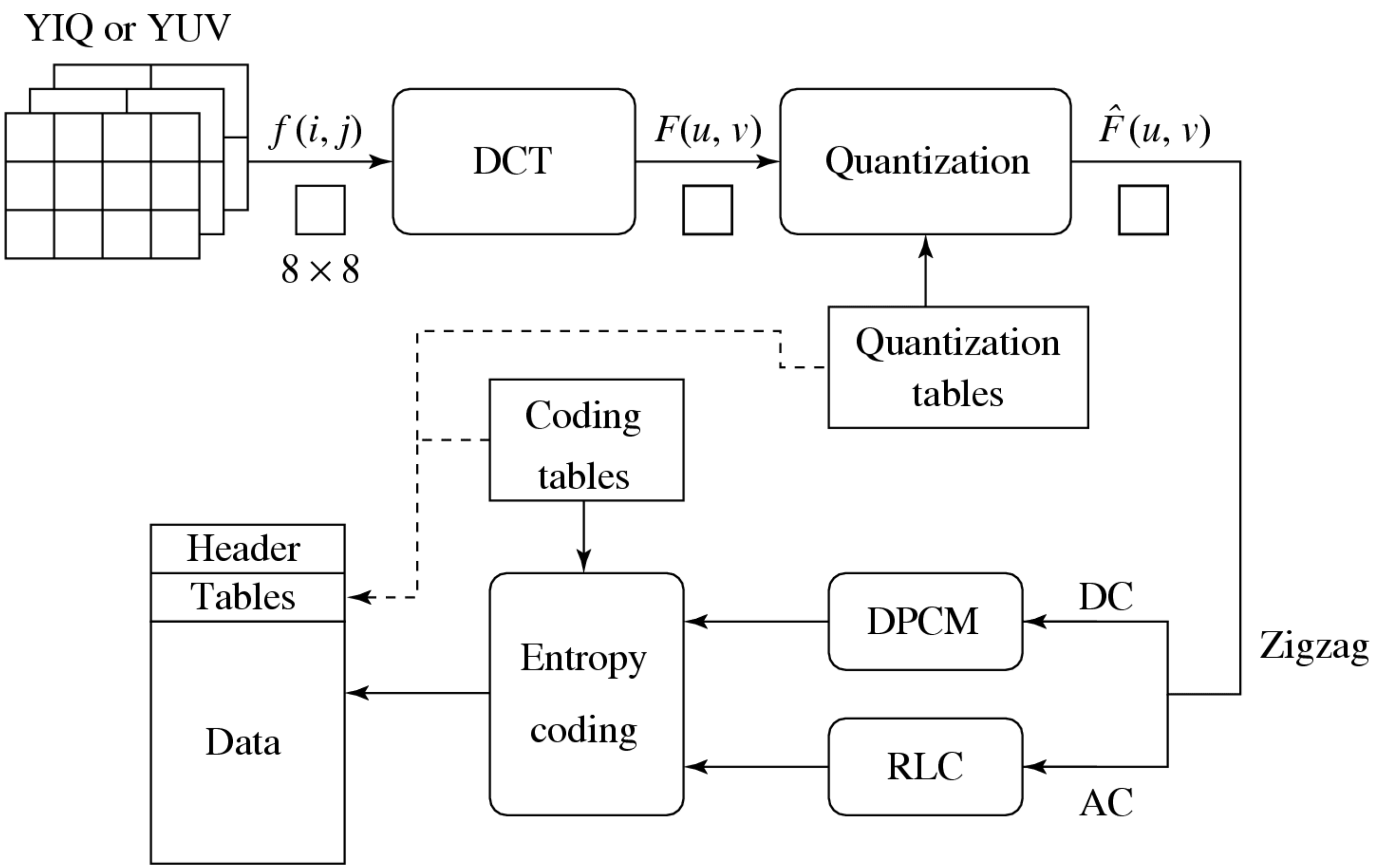
JPEG image compression

JPEG is an image compression standard that was developed by the "**J**oint **P**hotographic **E**xperts **G**roup".

JPEG was formally accepted as an international standard in 1992.

JPEG is a **lossy** image compression method.

Block diagram for JPEG encoder



Main Steps of JPEG

1. Transform **RGB to YIQ or YUV** and subsample color (skip this step for grayscale image)
2. Discrete Cosine Transform (**DCT**) of each 8*8 pixel array
3. **Quantization** using a table or using a constant
4. **Zig-zag** scan to exploit redundancy
5. Differential Pulse Code Modulation (**DPCM**) on the DC component
6. Run-length encoding (**RLC**) of the AC components
7. Entropy coding of the final output

Discrete Cosine Transform

DCT is a mathematical process related to Fourier Transform. It changes **spatial intensity values** to **spatial frequency values**.

Roughly arranges values from lowest frequency to highest frequency:

- **Lowest** frequencies represent **coarse** details.
- **Highest** frequencies represent **fine** details.
- Some high frequency parts can be dropped.

DCT exploits features of the human eye.

- The eye is unable to perceive brightness levels above or below a certain threshold.
- Gentle gradations of brightness of colour are more important.

1-D DCT

The one-dimensional transform is defined by

$$F(u) = c(u) \sum_{i=0}^{N-1} f(i) \cos\left(\frac{\pi(2i+1)}{2N} u\right)$$

$f(i)$: array of N original values (the luminance values of the pixel matrix)

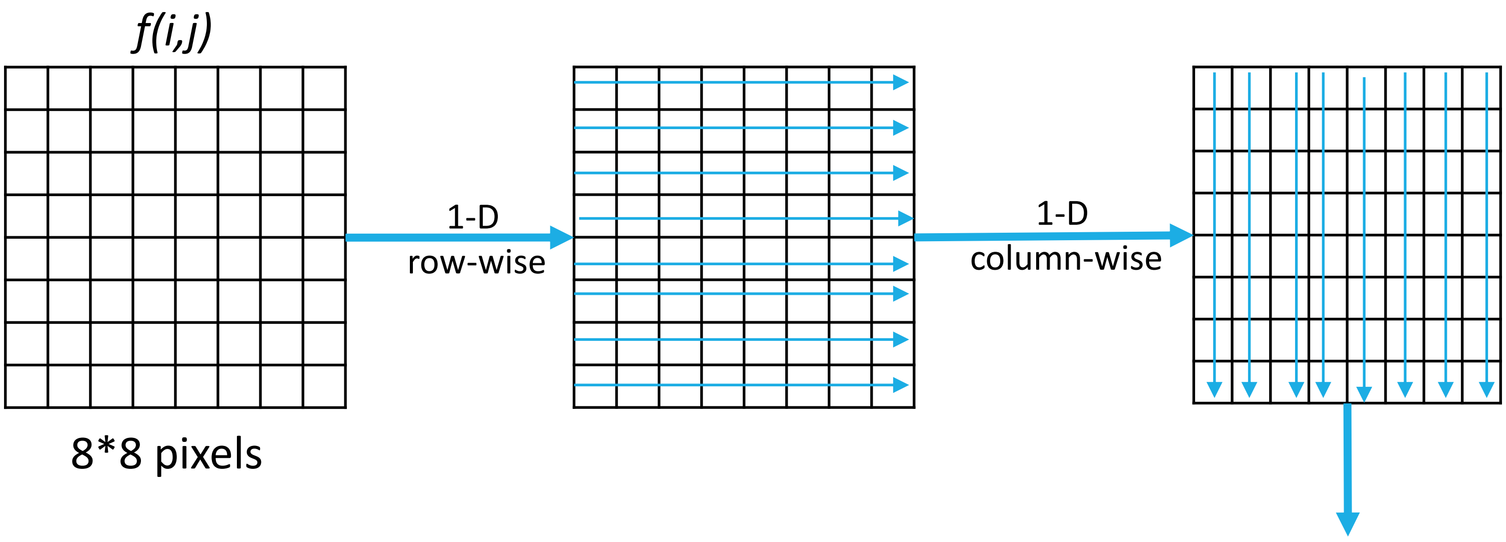
$F(u)$: array of N transformed values (the frequency coeffcients)

c: scaling factor for the coefficients given by $c(0) = \sqrt{1/N}, c(u) = \sqrt{2/N}$

Computing a 2-D DCT for images

Each image is divided into blocks of 8 × 8 pixels.

The 2D DCT is applied to each block image *f(i, j)*, with output being the DCT coefficients *F(u, v)* for each block.



F(0,0) is called DC component and
the rest of F(i,j) are called AC components.

DCT on image blocks

Using blocks has the effect of isolating each block from its neighboring context.
This is why JPEG images look choppy ("blocky") when a high *compression ratio* is specified by the user.

Quantization

The quantization step is the main source for loss in JPEG compression.

The quantized DCT coefficients is:

$$\hat{F}(u,v) = round\left(\frac{F(u,v)}{Q(u,v)}\right)$$

Where $F(u, v)$ is the DCT coefficients and $Q(u, v)$ is a quantization matrix entry.

- The entries of $Q(u, v)$ tend to have larger values towards the lower right corner.
- This aims to introduce more loss at the higher spatial frequencies.

Quantization

The Luminance(Y) Quantization Table

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

The Chrominance Quantization Table

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

The tables show the default Q(u, v) values obtained from psychophysical studies with the goal of maximizing the compression ratio while minimizing perceptual losses in JPEG images.

Quantization

An 8 × 8 block from the Y image of 'Lena'

Original block

200	202	189	188	189	175	175	175
200	203	198	188	189	182	178	175
203	200	200	195	200	187	185	175
200	200	200	200	197	187	187	187
200	205	200	200	195	188	187	175
200	200	200	200	200	190	187	175
205	200	199	200	191	187	187	175
210	200	200	200	188	185	187	186

*f(i, j)-the 8*8 block*

DCT coefficients of Original block

515	65	-12	4	1	2	-8	5
-16	3	2	0	0	-11	-2	3
-12	6	11	-1	3	0	1	-2
-8	3	-4	2	-2	-3	-5	-2
0	-2	7	-5	4	0	-1	-4
0	-3	-1	0	4	1	-1	0
3	-2	-3	3	3	-1	-1	3
-2	5	-2	4	-2	2	-3	0

F(u, v)

JPEG compression for a smooth image block.

Quantization

Quantized DCT coefficients

32	6	−1	0	0	0	0	0
−1	0	0	0	0	0	0	0
−1	0	1	0	0	0	0	0
−1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$\hat{F}(u, v)$

Dequantized DCT coefficients

512	66	−10	0	0	0	0	0
−12	0	0	0	0	0	0	0
−14	0	16	0	0	0	0	0
−14	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$\tilde{F}(u, v)$

IDCT of quantized DCT coefficients

199	196	191	186	182	178	177	176
201	199	196	192	188	183	180	178
203	203	202	200	195	189	183	180
202	203	204	203	198	191	183	179
200	201	202	201	196	189	182	177
200	200	199	197	192	186	181	177
204	202	199	195	190	186	183	181
207	204	200	194	190	187	185	184

$\tilde{f}(i, j)$

Difference btwn original and jpeg block

1	6	−2	2	7	−3	−2	−1
−1	4	2	−4	1	−1	−2	−3
0	−3	−2	−5	5	−2	2	−5
−2	−3	−4	−3	−1	−4	4	8
0	4	−2	−1	−1	−1	5	−2
0	0	1	3	8	4	6	−2
1	−2	0	5	1	1	4	−6
3	−4	0	6	−2	−2	2	2

$\epsilon(i, j) = f(i, j) - \tilde{f}(i, j)$

Fig. (cont'd): JPEG compression for a smooth image block.

Quantization

Another 8 × 8 block from the Y image of 'Lena'

Original block

70	70	100	70	87	87	150	187
85	100	96	79	87	154	87	113
100	85	116	79	70	87	86	196
136	69	87	200	79	71	117	96
161	70	87	200	103	71	96	113
161	123	147	133	113	113	85	161
146	147	175	100	103	103	163	187
156	146	189	70	113	161	163	197

$f(i, j)$

DCT coefficients of Original block

-80	-40	89	-73	44	32	53	-3
-135	-59	-26	6	14	-3	-13	-28
47	-76	66	-3	-108	-78	33	59
-2	10	-18	0	33	11	-21	1
-1	-9	-22	8	32	65	-36	-1
5	-20	28	-46	3	24	-30	24
6	-20	37	-28	12	-35	33	17
-5	-23	33	-30	17	-5	-4	20

$F(u, v)$

JPEG compression for a textured image block.

Quantization

Quantized DCT coefficients

−5	−4	9	−5	2	1	1	0
−11	−5	−2	0	1	0	0	−1
3	−6	4	0	−3	−1	0	1
0	1	−1	0	1	0	0	0
0	0	−1	0	0	1	0	0
0	−1	1	−1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$\hat{F}(u, v)$

Dequantized DCT coefficients

−80	−44	90	−80	48	40	51	0
−132	−60	−28	0	26	0	0	−55
42	−78	64	0	−120	−57	0	56
0	17	−22	0	51	0	0	0
0	0	−37	0	0	109	0	0
0	−35	55	−64	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$\tilde{F}(u, v)$

IDCT of quantized DCT coefficients

70	60	106	94	62	103	146	176
85	101	85	75	102	127	93	144
98	99	92	102	74	98	89	167
132	53	111	180	55	70	106	145
173	57	114	207	111	89	84	90
164	123	131	135	133	92	85	162
141	159	169	73	106	101	149	224
150	141	195	79	107	147	210	153

$\tilde{f}(i, j)$

Difference btwn original and jpeg block

0	10	−6	−24	25	−16	4	11
0	−1	11	4	−15	27	−6	−31
2	−14	24	−23	−4	−11	−3	29
4	16	−24	20	24	1	11	−49
−12	13	−27	−7	−8	−18	12	23
−3	0	16	−2	−20	21	0	−1
5	−12	6	27	−3	−2	14	−37
6	5	−6	−9	6	14	−47	44

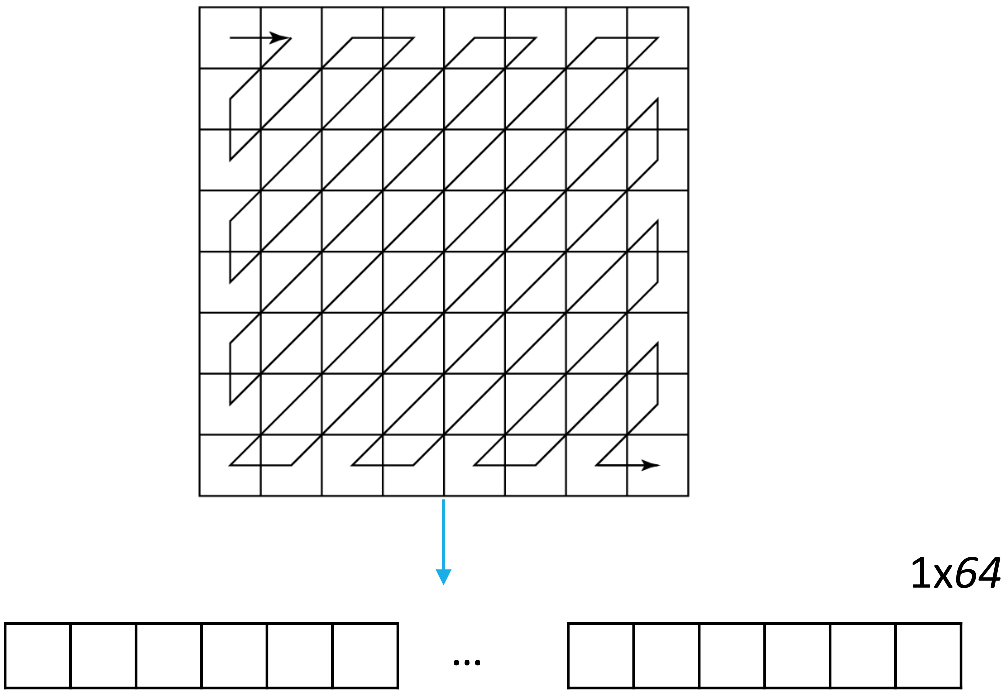
$\epsilon(i, j) = f(i, j) - \tilde{f}(i, j)$

Fig. (cont'd): JPEG compression for a textured image block.

Zig-zag scan

To group low frequency coefficients in top of the vector.

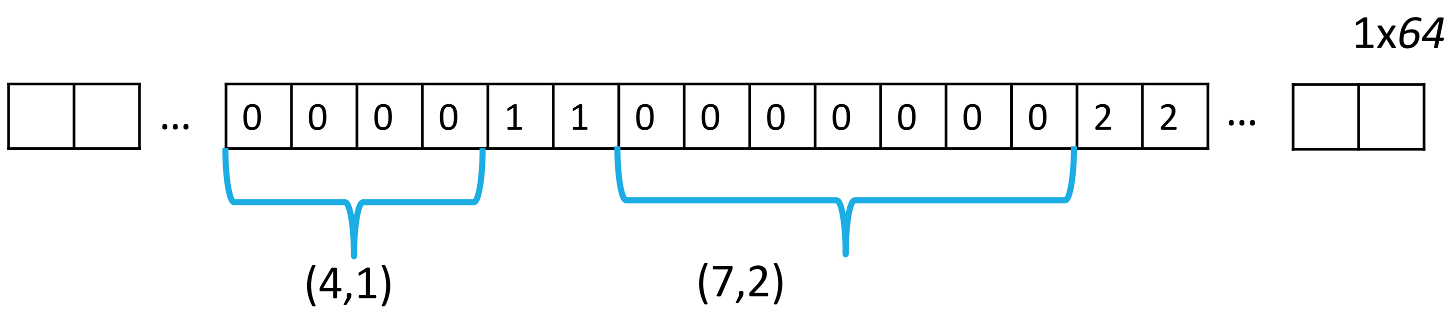
Map each 8x8 block into a 1x*64-vector*.



RLC on AC coefficients

The 1x64 vectors has **a lot of zeros** in them, more zeros towards to end of the vector.

Encode a series of 0s as a (*skip, value*) pair, where *skip* is the number of zeros and *value* is the next non-zero value

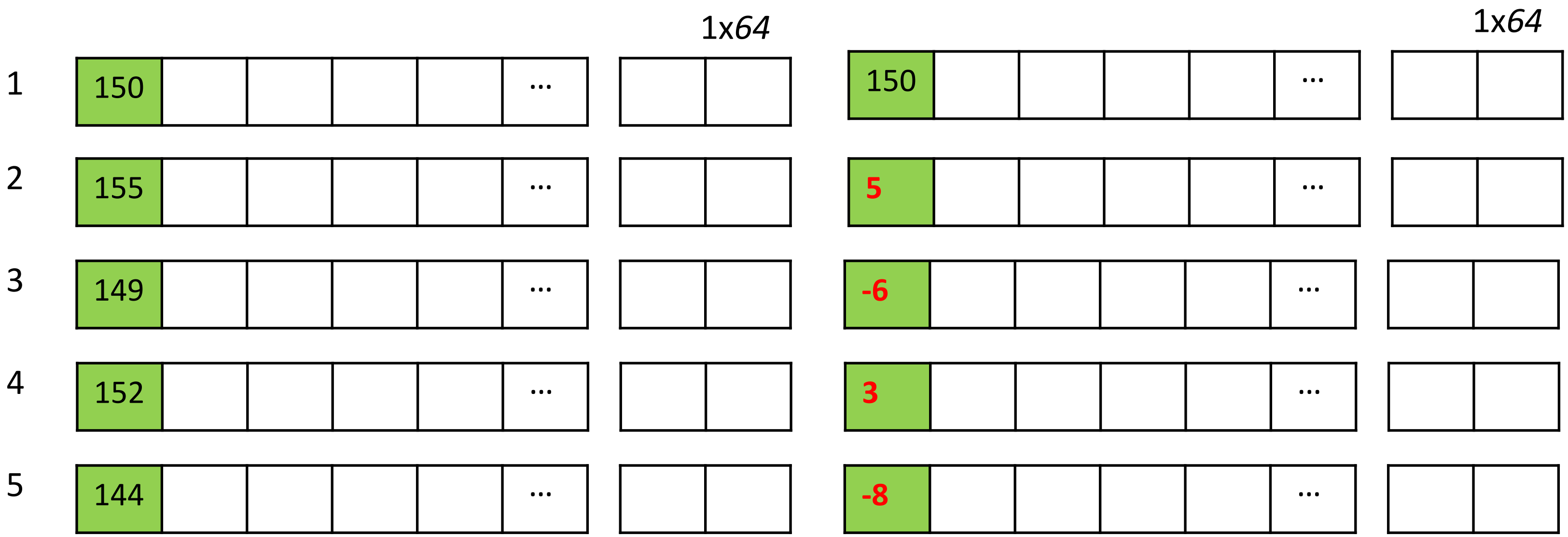


Example: RLC coding on an 8x8 block

DPCM on DC coefficients

Encode the **difference between the current and pervious** 8x8 block.

The DC component value in each 8x8 block is large and varies across the blocks but is often close to that in the previous block.



DPCM example

Entropy Coding

Entropy coding for both **AC and DC** components.

DPCM coded DC coefficient coded as (*size, value*)

- *Size* indicates how many bits are needed for representing the coefficient.
- Code of *value* is derived from the following table.

SIZE	Value	Code
1	-1, 1	0,1
2	-3, -2, 2, 3	00,01,10,11
3	-7..-4, 4..7	000,…,011,100,…,111
4	-15..-8, 8..15	0000,…,0111,1000,…,1111
.	.	…
10	-1023..-512, 512..1023	…

Example: 150,5,-6,3,8→(8,10010110), (3,101), (3,001), (2,11), (4,0111)

Entropy Coding

Size is Huffman coded since smaller *Sizes* occur much more often.

Value is not Huffman coded, its value can change widely so Huffman coding has no appreciable benefit.

SIZE	Code length	Code
0	2	00
1	3	010
2	3	011
3	3	100
4	3	101
5	3	110
6	4	1110
7	5	11110
8	6	111110
9	7	1111110
10	8	11111110
11	9	111111110

Example: 150,5,-6,3,8→

(**8**,10010110), (**3**,101), (**3**,001), (**2**,11), (**4**,0111)→

(**111110**,
10010110),(**100**,101),(**100**,001),(**011**,11),(**101**,0111)→

11111010010110 100101 100001 01111 1010111

4 JPEG Modes

Sequential mode

- each gray level or colour image is encoded in a single left-to-right, top-to-bottom scan
- baseline sequential mode (technique discussed so far)
  - based on DCT and quantization (the source of the loss)

Progressive Mode

Hierarchical Mode

Lossless mode

- predictive coding mechanism

Progressive Mode

Progressive JPEG delivers low quality versions of the image quickly, followed by higher quality passes.

- **1. Spectral selection**: Takes advantage of the “spectral” (spatial frequency spectrum) characteristics of the DCT coefficients: higher AC components provide detail information.

- Scan 1: Encode DC and first few AC components, e.g., AC1, AC2.
- Scan 2: Encode a few more AC components, e.g., AC3, AC4, AC5.
- ...
- Scan k: Encode the last few ACs, e.g., AC61, AC62, AC63.

Progressive Mode

- 2. **Successive approximation**: Instead of gradually encoding spectral bands, all DCT coefficients are encoded simultaneously but with their most significant bits (MSBs) first.

- Scan 1: Encode the first few MSBs, e.g., Bits 7, 6, 5, 4.
- Scan 2: Encode a few more less significant bits, e.g., Bit 3.
- ...
- Scan m: Encode the least significant bit (LSB), Bit 0.

Hierarchical Mode

Hierarchical JPEG encodes the image in a hierarchy of several different resolutions.

It enables the decoder at receiver to chose the optimum resolution depending on the its capabilities.

The encoded image at the lowest resolution is basically a compressed low-pass filtered image, whereas the images at successively higher resolutions provide additional details (differences from the lower resolution images).

Similar to Progressive JPEG, the Hierarchical JPEG images can be transmitted in multiple passes progressively improving quality.

Hierarchical Mode: 2-level

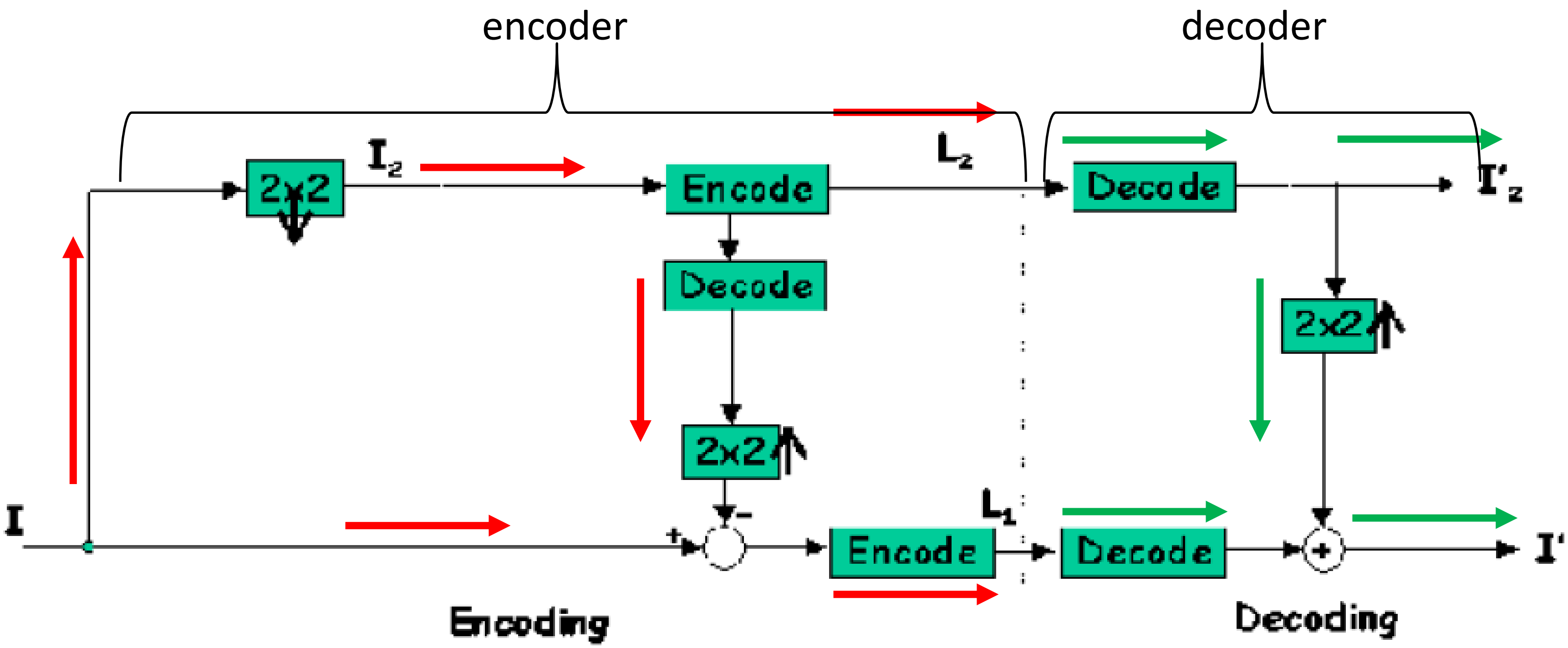


Figure 1. Two-level Hierarchical mode Encoder/Decoder

A Glance at the JPEG Bitstream

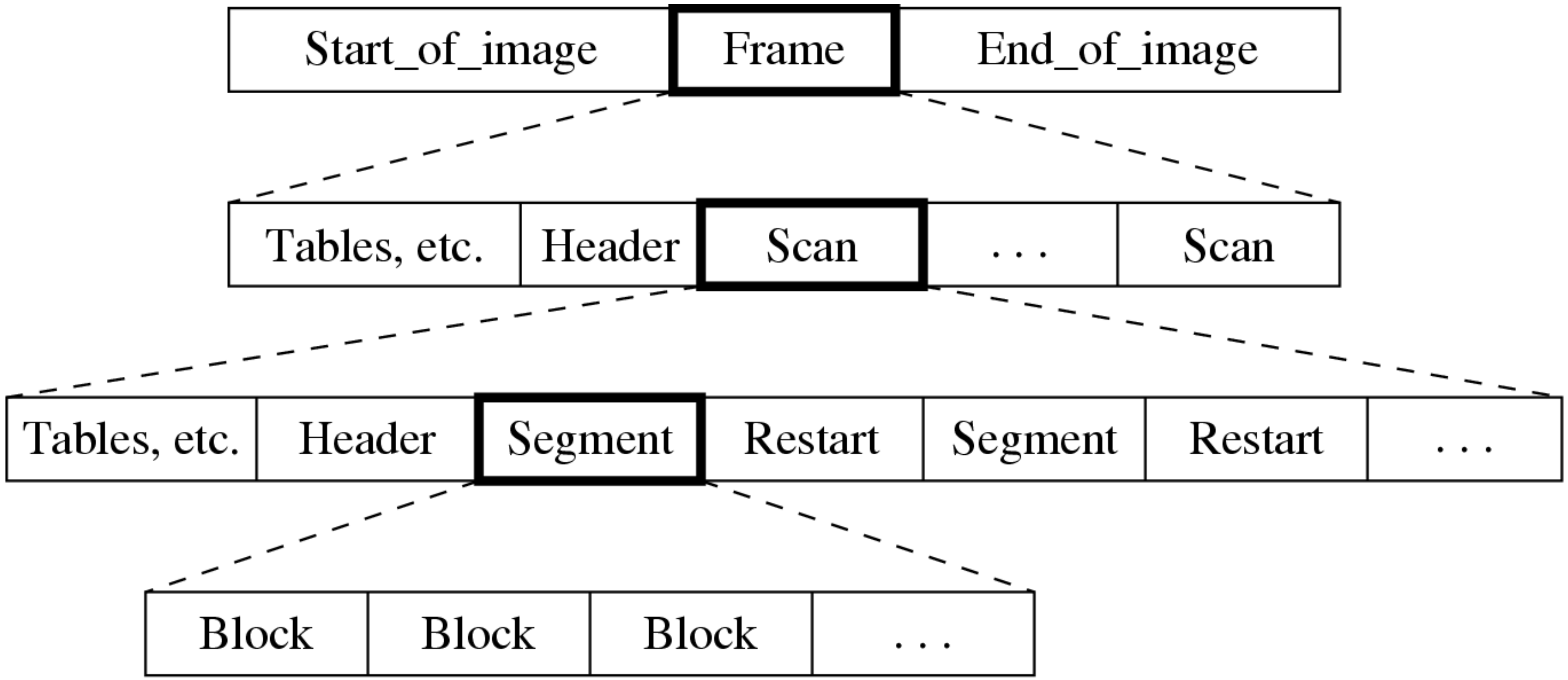


Fig. Hierarchical view of the JPEG bitstream.

The JPEG2000 Standard

JPEG 2000 (JP2) is an image compression standard and coding system. It was created by the Joint Photographic Experts Group committee in 2000.

The standardized filename extension is .jp2.

JPEG 2000 code streams are regions of interest that offer several mechanisms to support **spatial random access** or **region of interest** access at varying degrees of granularity. It is possible to store different parts of the same picture using different quality.

The JPEG2000 Standard

Design Goals

- To provide a better rate-distortion trade-off and improved subjective image quality.
- To provide additional functionalities lacking in the current JPEG standard.

The JPEG2000 standard addresses the following problems:

Lossless and Lossy Compression

- Currently no standard that can provide superior lossless compression and lossy compression in a single bitstream.

The JPEG2000 Standard

Low Bit-rate Compression:

- The current JPEG offers excellent rate-distortion performance in mid and high bit-rates.
- at bit-rates below 0.25 bpp, subjective distortion becomes unacceptable.

Large Images:

- The new standard will allow image resolutions greater than 64K by 64K without tiling. It can handle image size up to $2^{32} - 1$.

Single Decompression Architecture

- The current JPEG has 44 modes, many of which are application specific and not used by the majority of JPEG decoders.
- Single common decompression architecture can provide greater interchange between applications.

The JPEG2000 Standard

Transmission in Noisy Environments

- The new standard will provide improved error resilience for transmission in noisy environments such as wireless networks and the Internet.

Progressive Transmission

- The new standard provides seamless quality and resolution scalability from low to high bit-rate. The target bit-rate and reconstruction resolution need not be known at the time of compression.

Region of Interest Coding

- The new standard allows the specification of Regions of Interest (ROI) which can be coded with superior quality than the rest of the image. One might like to code the face of a speaker with more quality than the surrounding furniture.

The JPEG2000 Standard

Computer Generated Imagery

- The current JPEG standard is optimized for natural imagery and does not perform well on computer generated imagery.

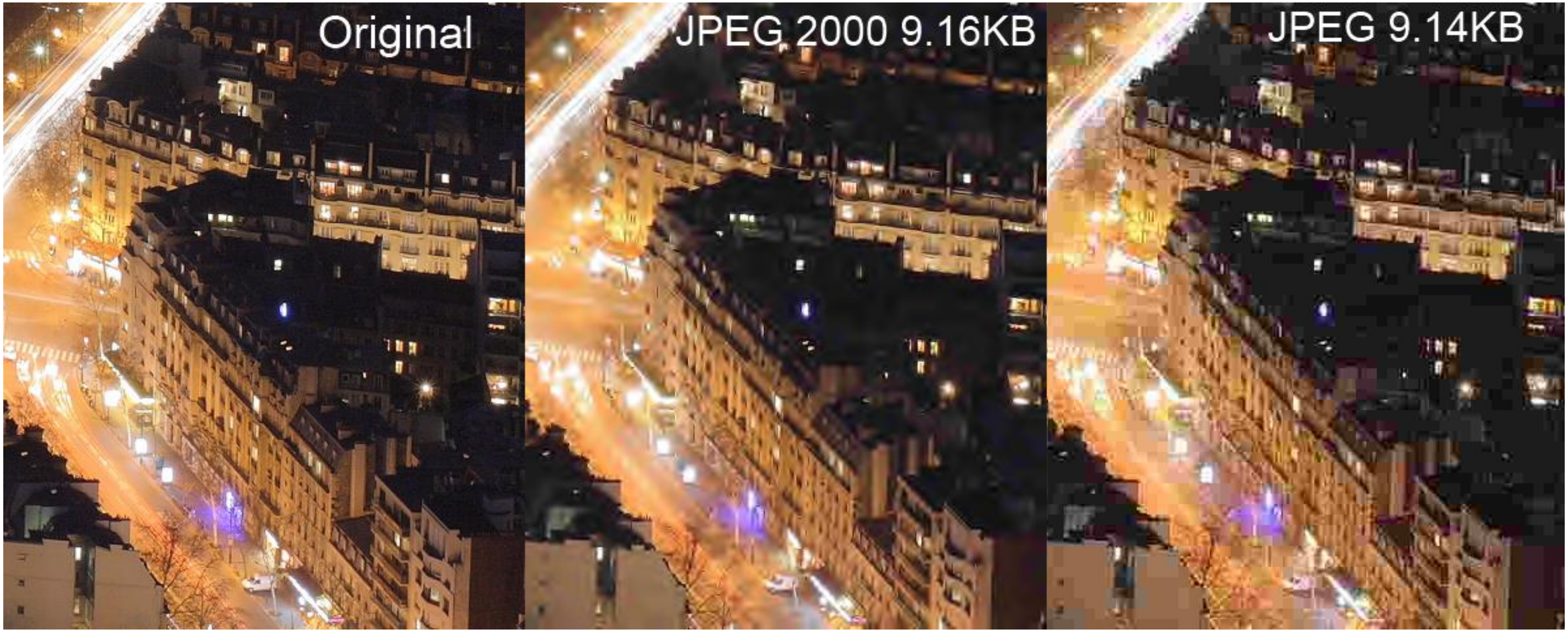
Compound Documents

- The new standard offers metadata mechanisms for incorporating additional non-image data as part of the file. This might be useful for including text along with imagery, as one important example.

JPEG2000 handle up to 256 channels of information whereas the current JPEG standard is only able to handle three color channels.

The JPEG2000 standard operates in two coding modes: DCT-based and wavelet-based.

JPEG vs. JPEG2000

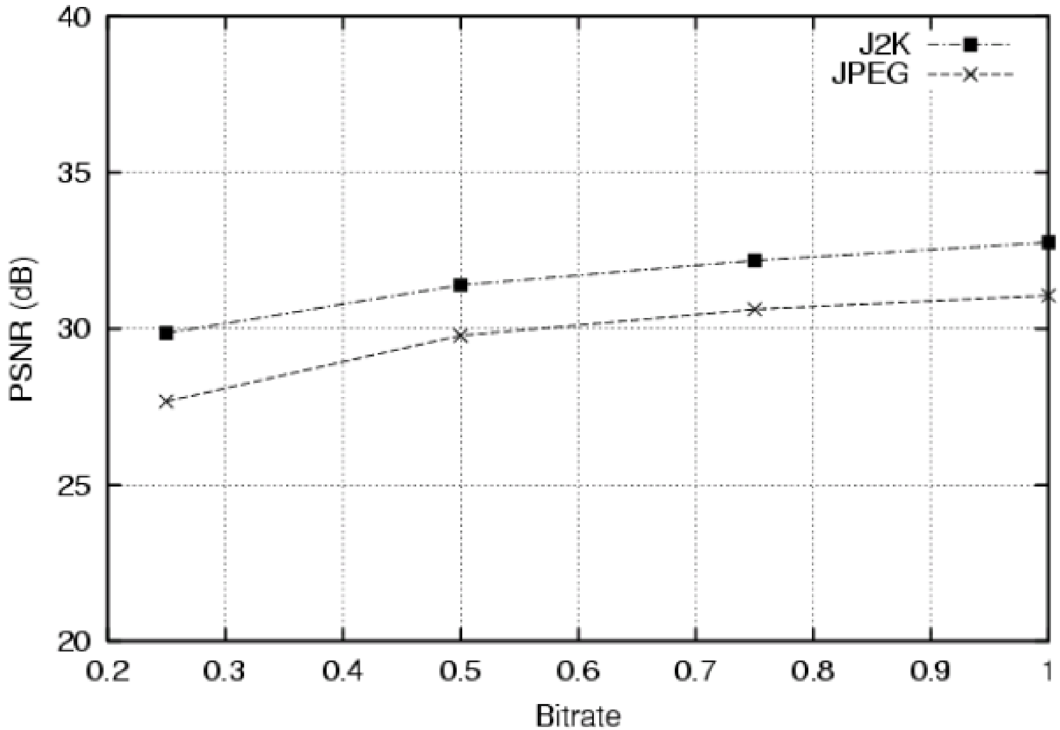


JPEG

JPEG 2000

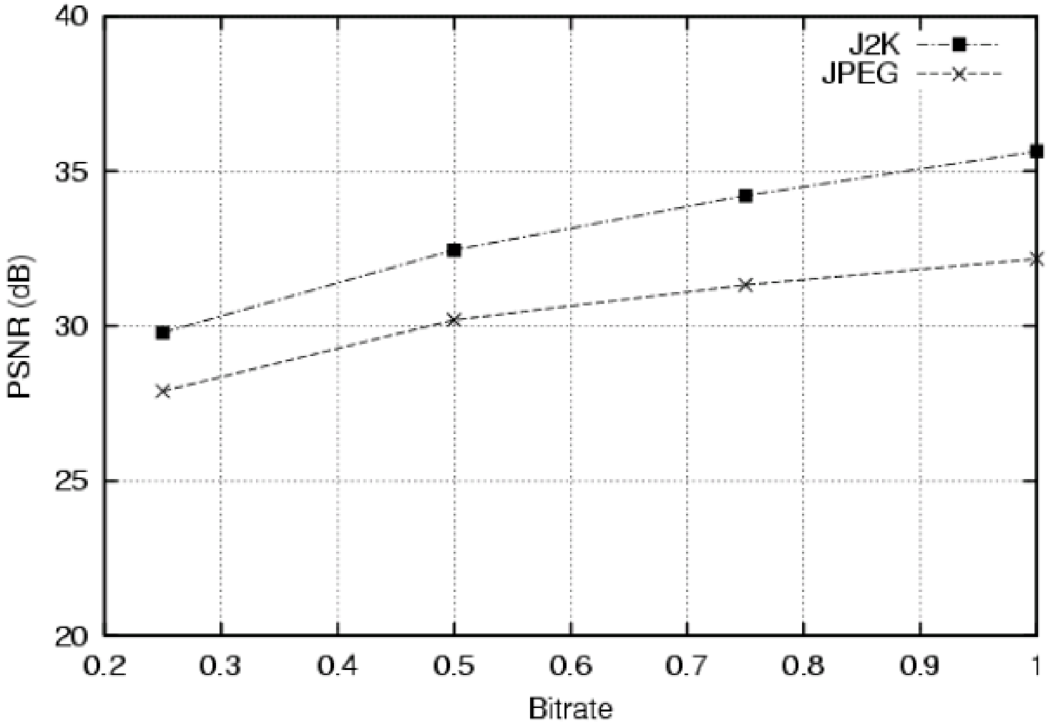
JPEG

JPEG 2000



(a)

Fig.: Performance comparison for JPEG and JPEG2000 on different image types. (a): Natural images.



(b)

Fig.: Performance comparison for JPEG and JPEG2000 on different image types. (b): Computer generated images.

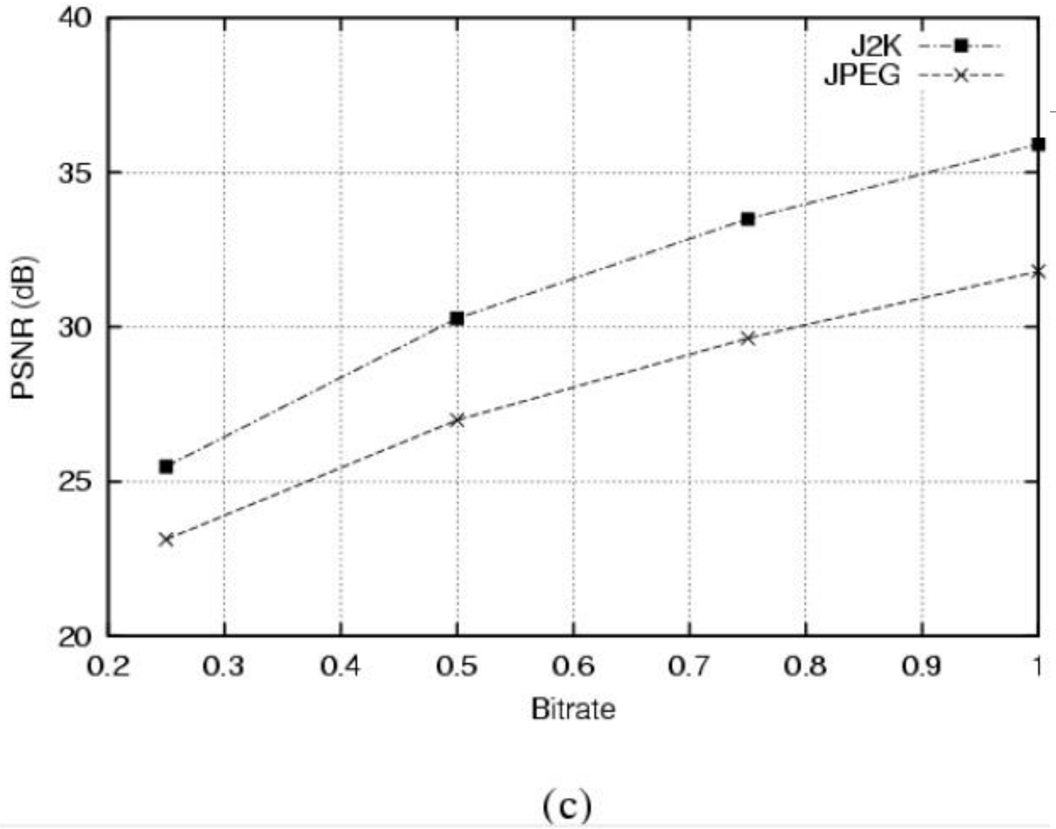
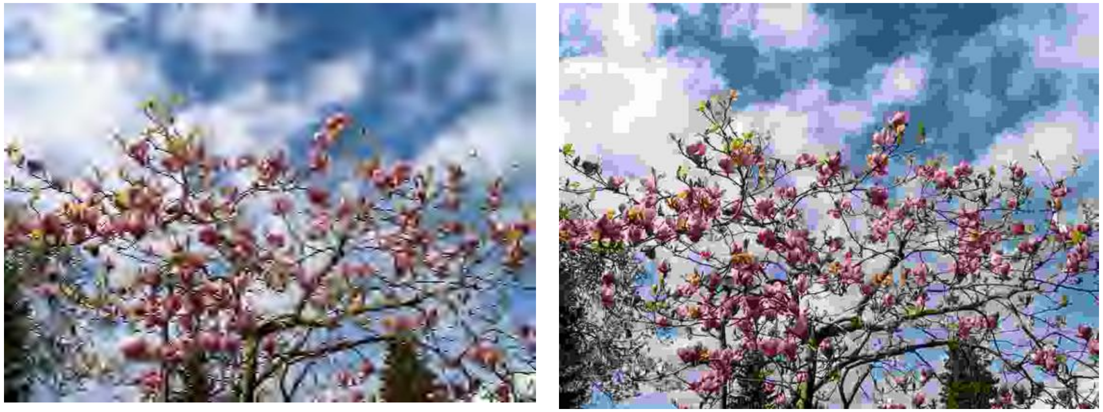


Fig. : Performance comparison for JPEG and JPEG2000 on different image types.
(c): Medical images.

(a)

Fig.: Comparison of JPEG and JPEG2000. (a) Original image.

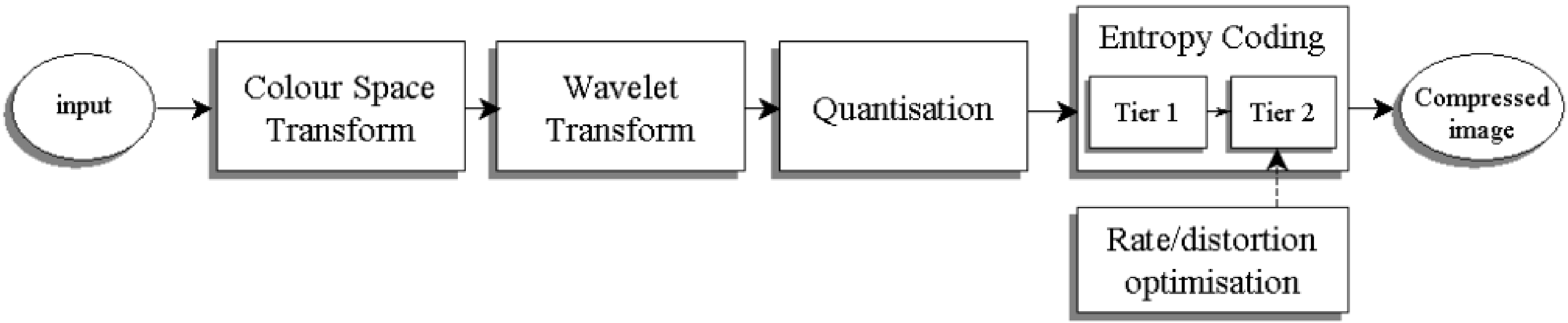
(b)



(c)

Fig. (Cont'd): Comparison of JPEG and JPEG2000. (b) JPEG (left) and JPEG2000 (right) images compressed at 0.75 bpp. (c) JPEG (left) and JPEG2000 (right) images compressed at 0.25 bpp.

JPEG2000

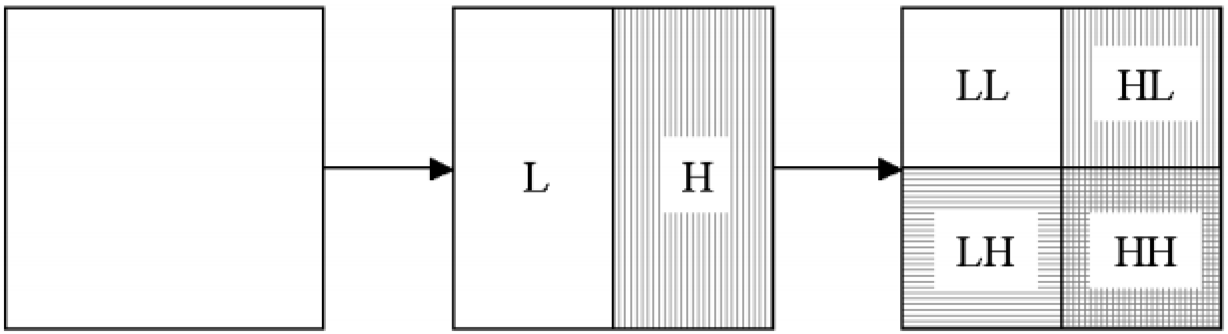


JPEG2000 image compression pipeline

Wavelet transform

Wavelet transform separates a tile into four sub-bands with different frequencies, source modelling can be tailored to each sub-band, LL, LH, HL, HH.

Wavelet transformed data usually exhibits a lower entropy and is thus more "compressible".



Wavelet transform-sub-band division and frequency filtering

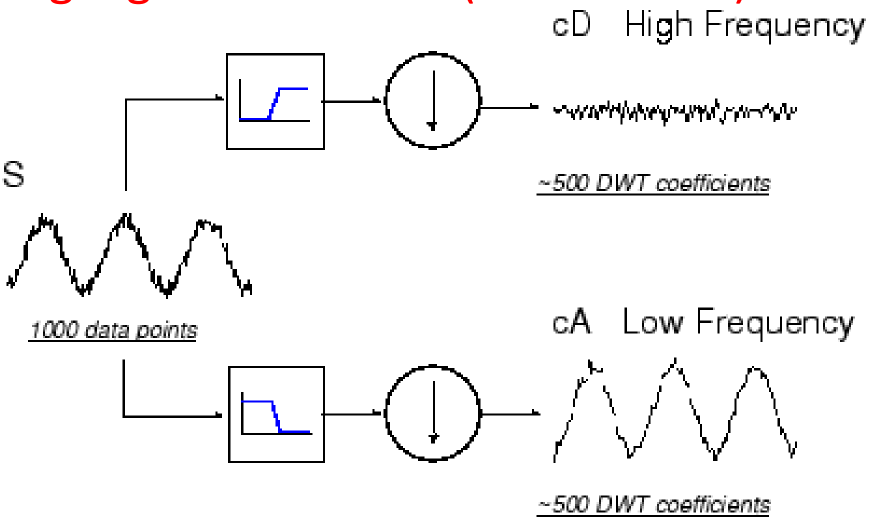
Discrete Wavelet Transform

$$DWT(m,n) = \langle f, \psi_{m,n} \rangle = 2^{-\frac{1}{2}} \sum_{-\infty}^{+\infty} f(k) \cdot \psi^*(2^{-m}k - n)$$

Where $\psi_{m,n}(k) = 2^{-\frac{m}{2}} \psi(2^{-m}k - n)$

The wavelet decomposition of a signal based on multi-resolution theory can be done using digital FIR filters. (**Filter Banks**)

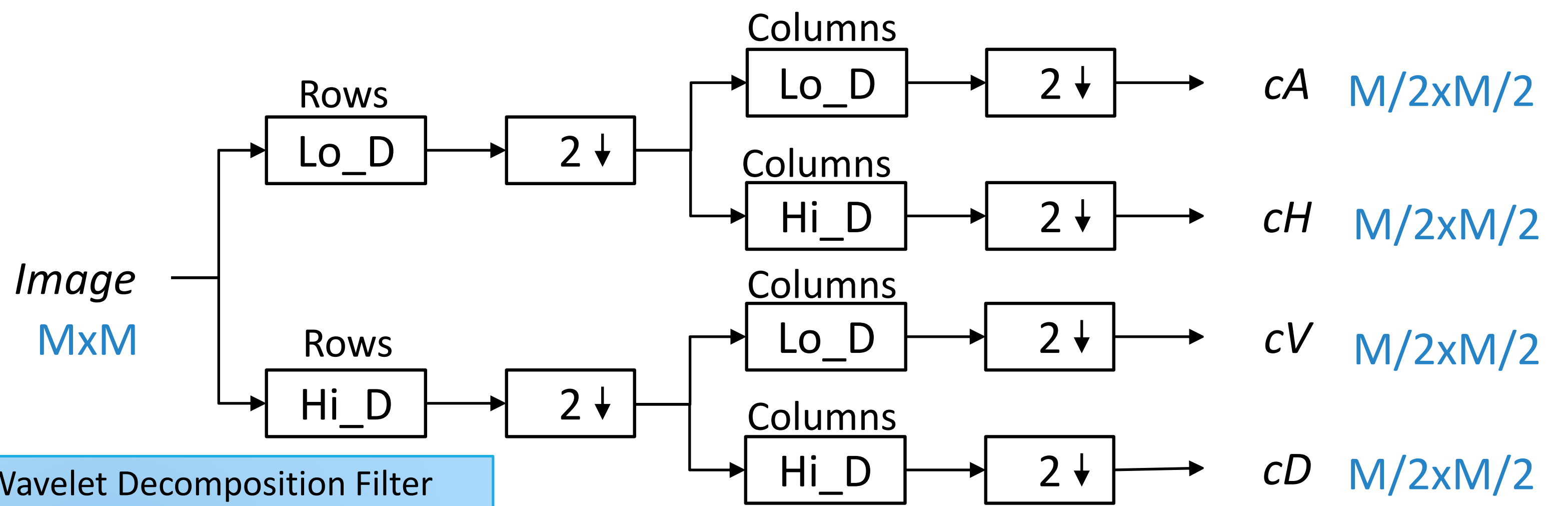
One level Decomposition for a 1D signal S



Filter banks: image decomposition

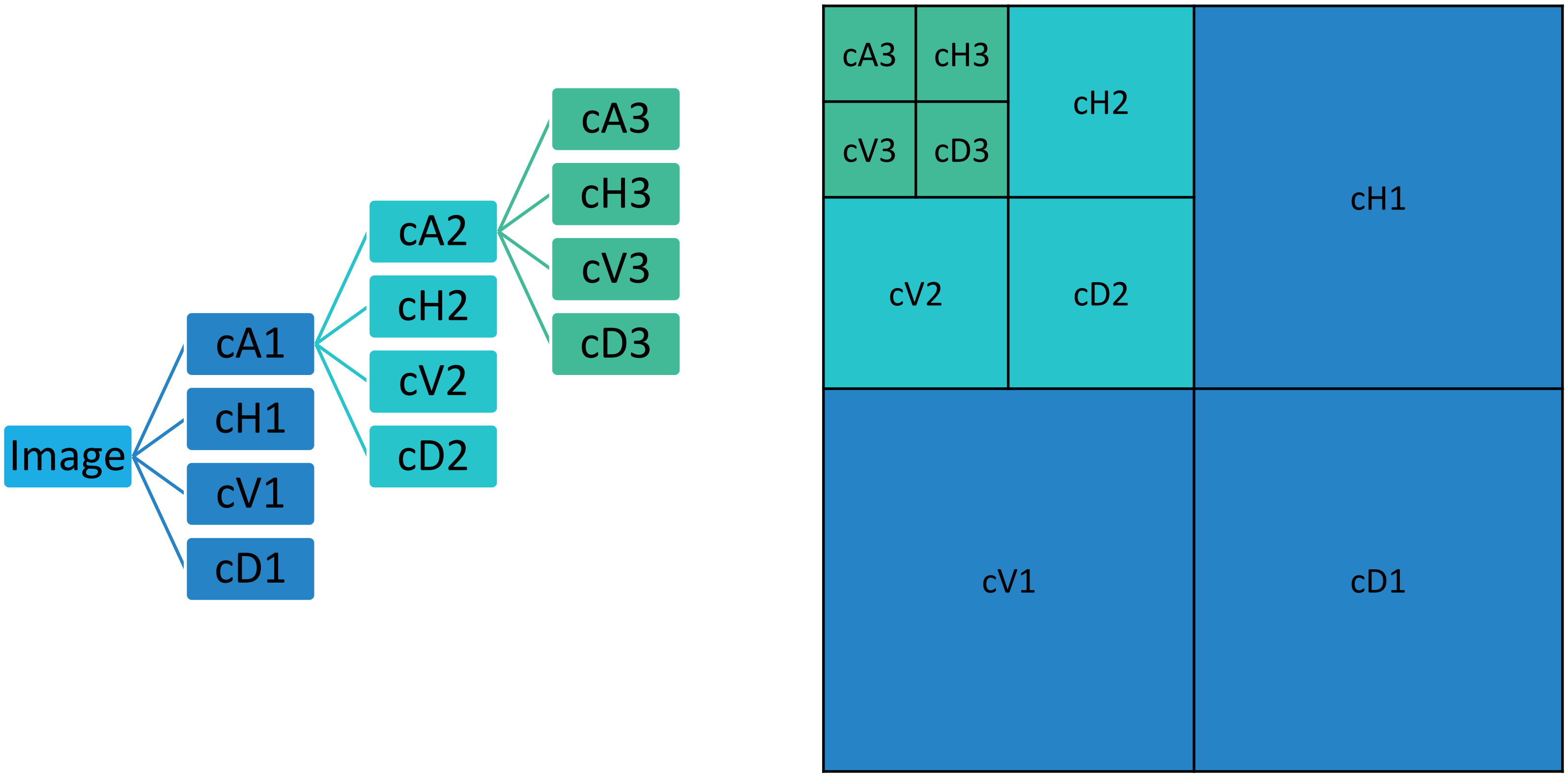
We can apply one dimensional transform to the rows and columns of the image successively as separable two-dimensional transform.

Advantages: low computation complexity!



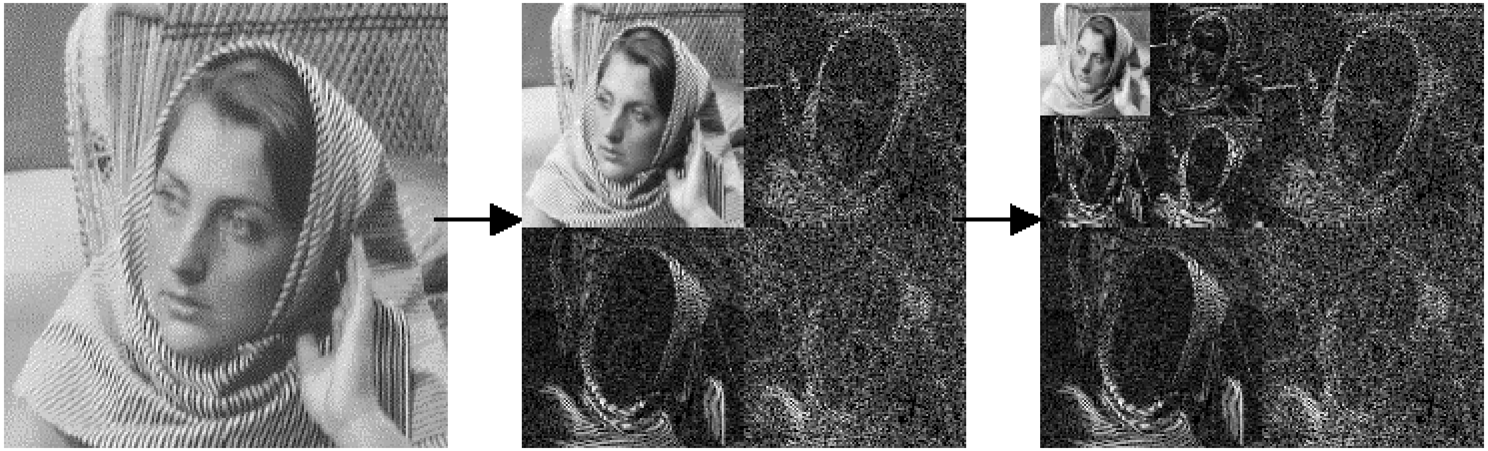
Lo_D: Low Pass Wavelet Decomposition Filter
Hi_D: High Pass Wavelet Decomposition Filter
2↓ : Down-sampling by 2
cA: Approximation Coefficients
cH: Horizontal Detailed Coefficients
cV: Vertical Detailed Coefficients
cD: Diagonal Detailed Coeffients

Multi-level decomposition



Wavelet transform

- Wavelet method transforms image as a whole (not subdivided into pixel blocks).
  - No blocking artifacts.

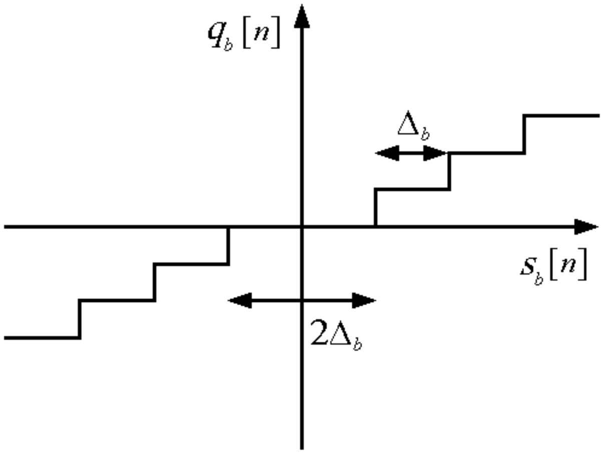


Wavelet transform-effect of frequency filtering on a grayscale image

Quantisation

Quantisation reduces the bit-depth of wavelet coefficients at the expenses of precision.

Two methods: scalar quantisation, and Trellis-coded quantisation



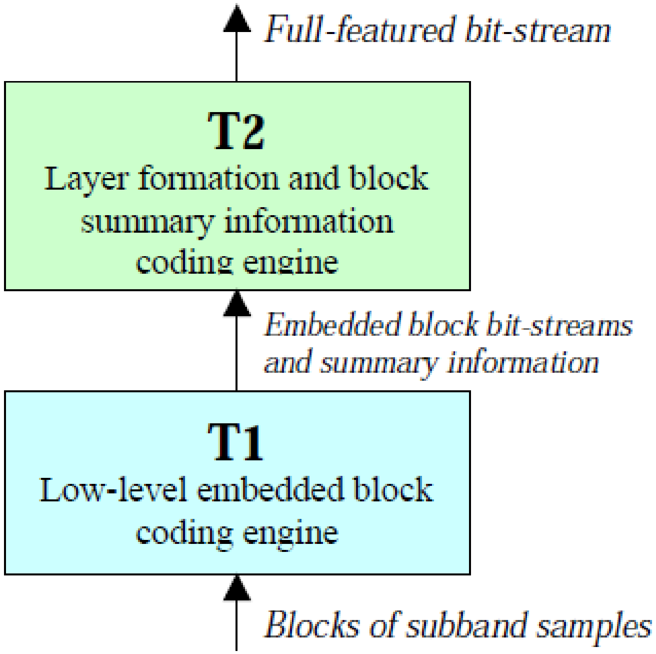
Scalar quantisation: a dead-zone is generated around zero.
$s_b[n]$ indicates the input signal while $q_b[n]$ is the output quantised signal.

EBCOT

Stands for ‘Embedded Block Coding with Optimised Truncation’

Two layers

- Tier 1 for source modelling and entropy coding
- Tier 2 for output stream generation

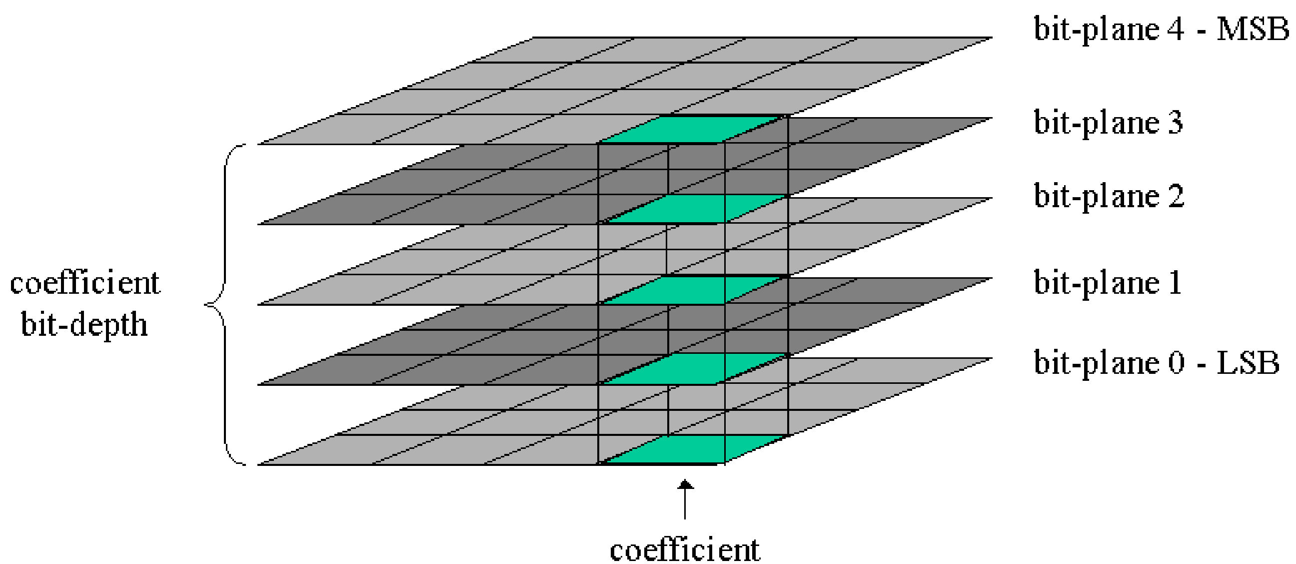


EBCOT Tier 1

Wavelet sub-bands are partitioned into small ‘*code-blocks*’.

Code-blocks in turn are coded by bit-planes.

- i.e. coefficient bits of the same order are coded together. The most significant bits are coded first, then low order bits are coded in descending order.



Bit-plane coding: an example of bit-plane coding of wavelet coefficients with bit-depth 5

EBCOT Tier 1

Each bit-plane gets encoded by three coding passes.

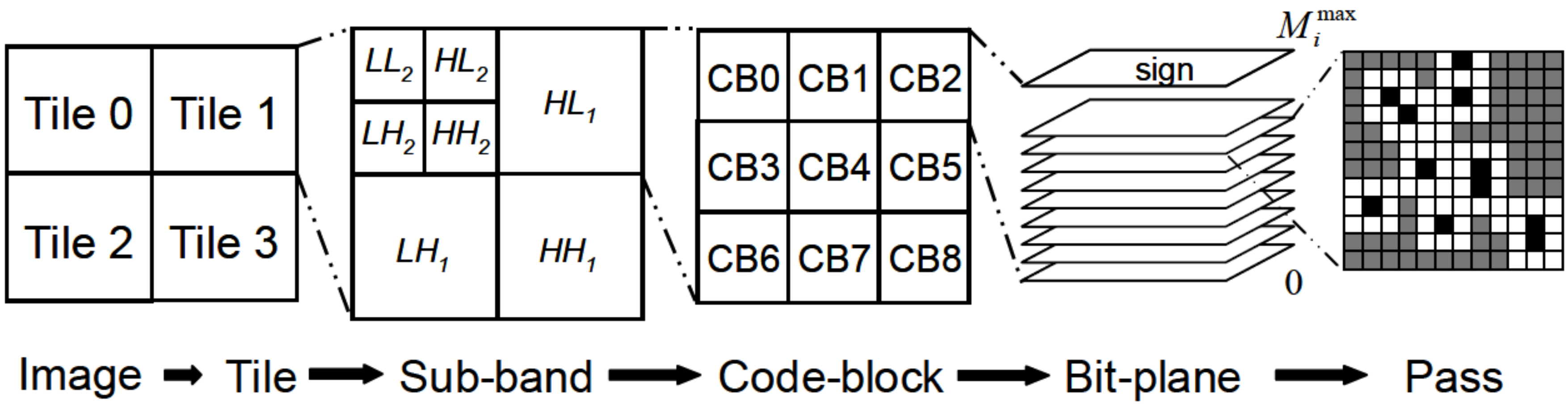
- 3 passes: Significance Propagation (SP), Magnitude Refinement (MR) and Cleanup (CU)
- Different coding schemes adopted in the coding passes.
  - SP: zero coding, sign coding
  - MR: magnitude refinement
  - CU: zero coding, sign coding, run length coding
  - Arithmetic coding is used to reduce complexity in ‘Lazy coding mode’.

EBCOT Tier 1

Each coding pass constitutes an atomic code unit, called '*chunk*'. *chunks* are grouped into *quality layers* and can be transmitted in any order.

Chunks constitute valid *truncation points*.

- i.e. the codeword can be truncated at the end of a chunk without compromising the correct reconstruction of the compressed data

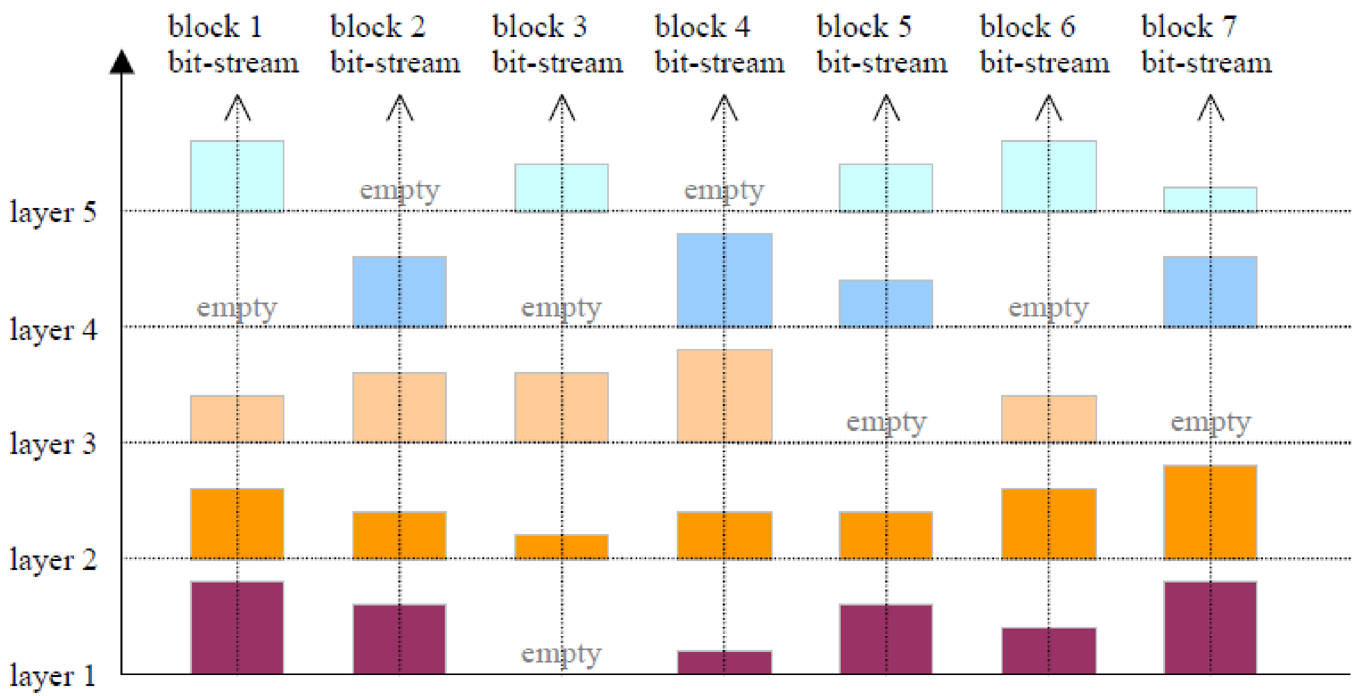


EBCOT Tier 1

5 layers and 6 code-blocks. Each rectangle is the chunks of a code-block. The data coded for a block is represented as the chunks lying on the

cc

tr



EBCOT Tier 2

To enable the user to break the stream up to the desired resolution and/or quality, or to visualise a particular area of the image, the compressed bitstream structure must be carefully organised.

The stream is partitioned into *packets* (header + stream itself)

- Headers contain information useful to retrieve codeword chunks in the output stream, e.g. whether any block of a given wavelet sub-band is included in the packet or not, codeword length, and so on.
- Tag Tree is adopted to exploit this redundancy in headers.

Region of Interest Coding in JPEG2000

Goal: Particular regions of the image may contain important information, thus should be coded with better quality than others.

Usually implemented using the MAXSHIFT method which scales up the coefficients within the ROI so that they are placed into higher bit-planes.

During the embedded coding process, the resulting bits are placed in front of the non-ROI part of the image. Therefore, given a reduced bit-rate, the ROI will be decoded and refined before the rest of the image.

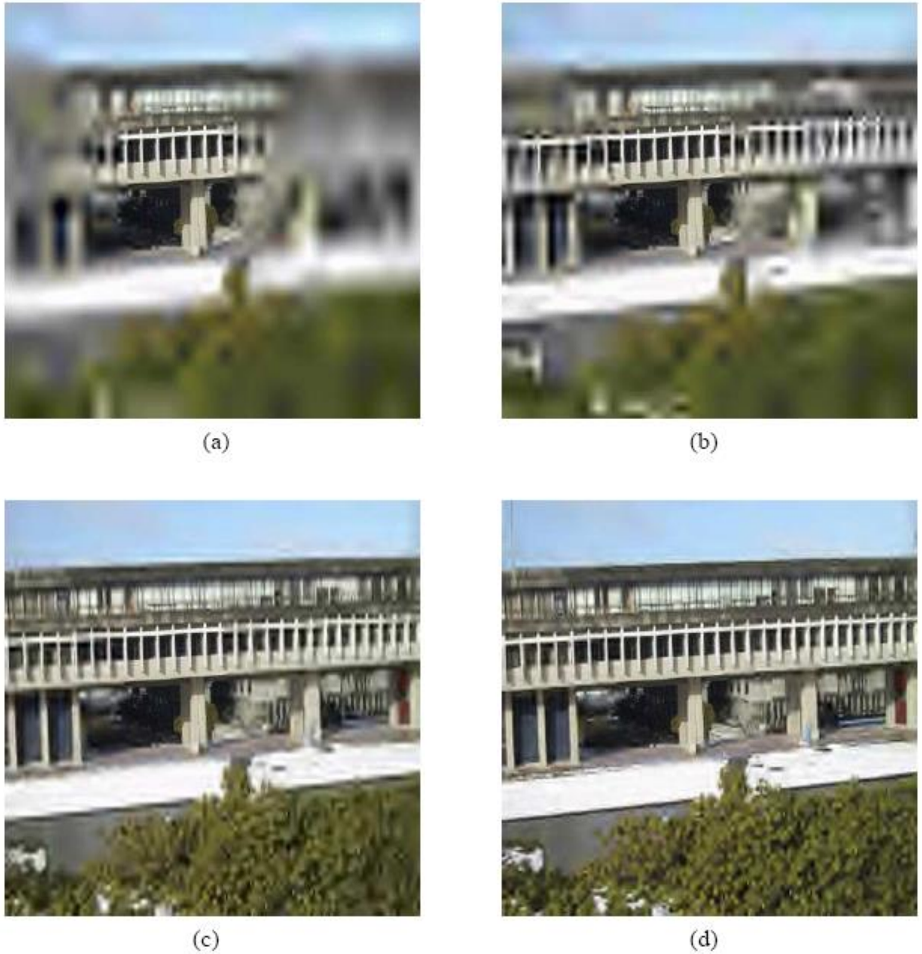


Fig.: Region of interest (ROI) coding of an image using a circularly shaped ROI. (a) 0.4 bpp, (b) 0.5 bpp, (c) 0.6bpp, and (d) 0.7 bpp.

Q&A