JPEG

Catania – 08/04/2008

Arcangelo Bruna

Advanced System Technology

JPEG – A Still Compression Standard

- JPEG is an acronym for "Joint Photographic Experts Group". (www.jpeg.org)

- The JPEG standard was developed for continuous-tone still image compression.

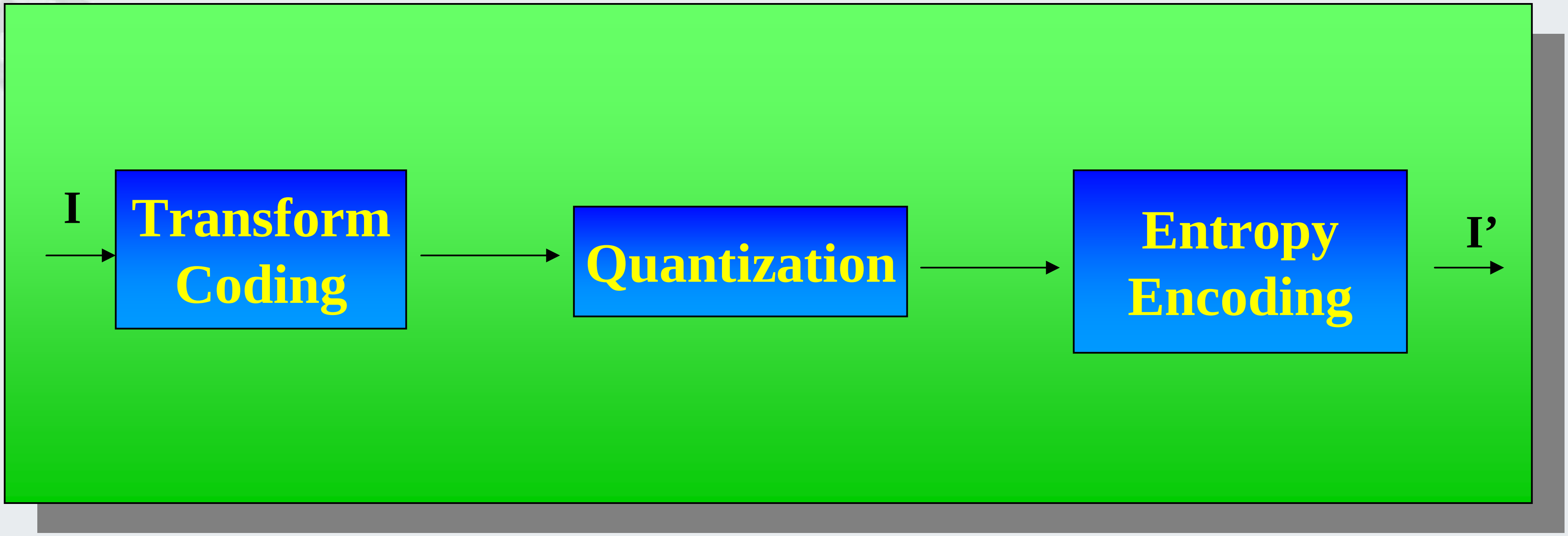
- In 1988, JPEG selected an adaptive DCT coding scheme as its backbone for the standard. The technical contents were further refined between 1988 and 1990.

- In 1991, a standard draft was sent to standard bodies for the official balloting process, and was adopted as an international standard in 1992 (ISO –International Standard Organization).

Standard

CCITT T. 4	Facsimile, Document Imaging.
CCITT T. 6	Facsimile, Document Imaging.
JPEG (JPEG2000)	Photographic Imaging.
JBIG	Facsimile, Document Imaging.
ITU H. 261	Teleconferencing, px64Kb/ s.
ITU H. 263	Improved H. 261, wide range of bitrates.
MPEG-1/2/4/7/ …	Video, Digital Storage Media (DSM), Video, HDTV, DSM, Audio- visual communications, Multimedia, Remote sensing, Audio/ Video content-based retrieval.

A Typical Compression System

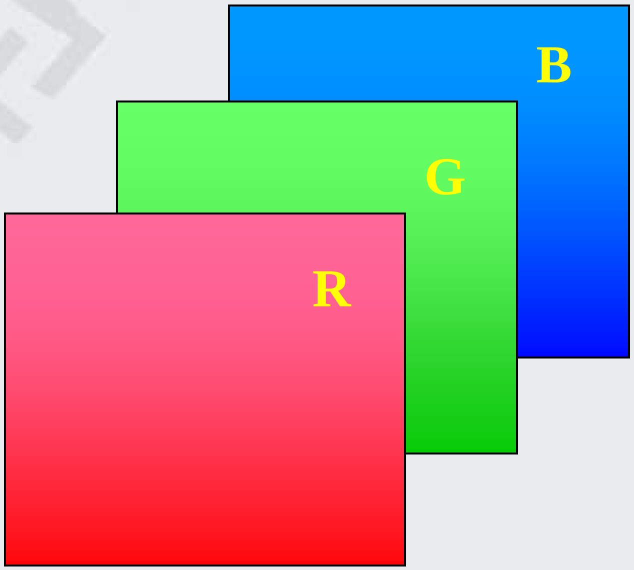


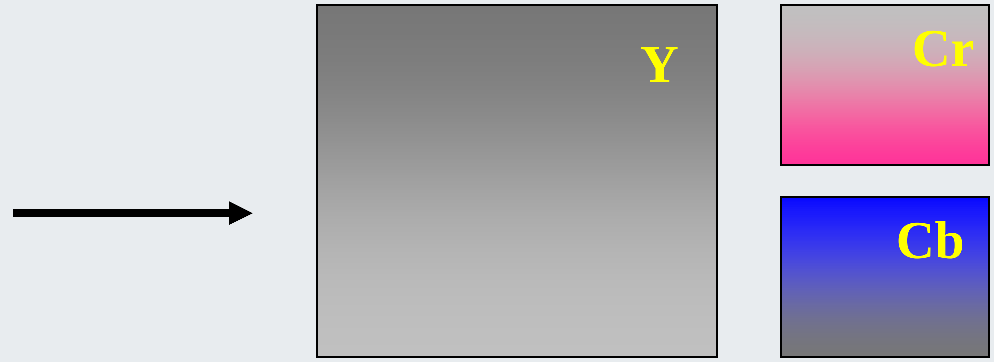
- *Most of the compression occurs in the quantization stage.*
- *Lossless compression/entropy coding, typically involves run-length coding combined with Huffman codes, further save bits in an invertible fashion.*

JPEG Baseline Encoding Process

- **Color Transform (RGB $\rightarrow$ YCbCr);**
- **Image Partition;**
- **Discrete Cosine Transform;**
- **Quantization;**
- **DC Coefficient Encoding;**
- **Zig-zag ordering of AC Coefficients;**
- **Entropy Coding.**

Color Transform





Y = 0.299 R + 0.587 G + 0.114 B

Cb = (B – Y)/2 + 0.5

Cr = (R – Y)/2 + 0.5

The human eye is more sensitive to luminance than to chrominance. Typically JPEG throw out $^3/_4$ of the chrominance information before any other compression takes place. This reduces the amount of information to be stored about the image by $^1/_2$. With all three components fully stored, 4 pixels needs 3 x 4 = 12 component values. If $^3/_4$ of two components are discarded we need 1 x 4 + 2 x 1 = 6 values.

Example:

RGB → YUV and subsampling

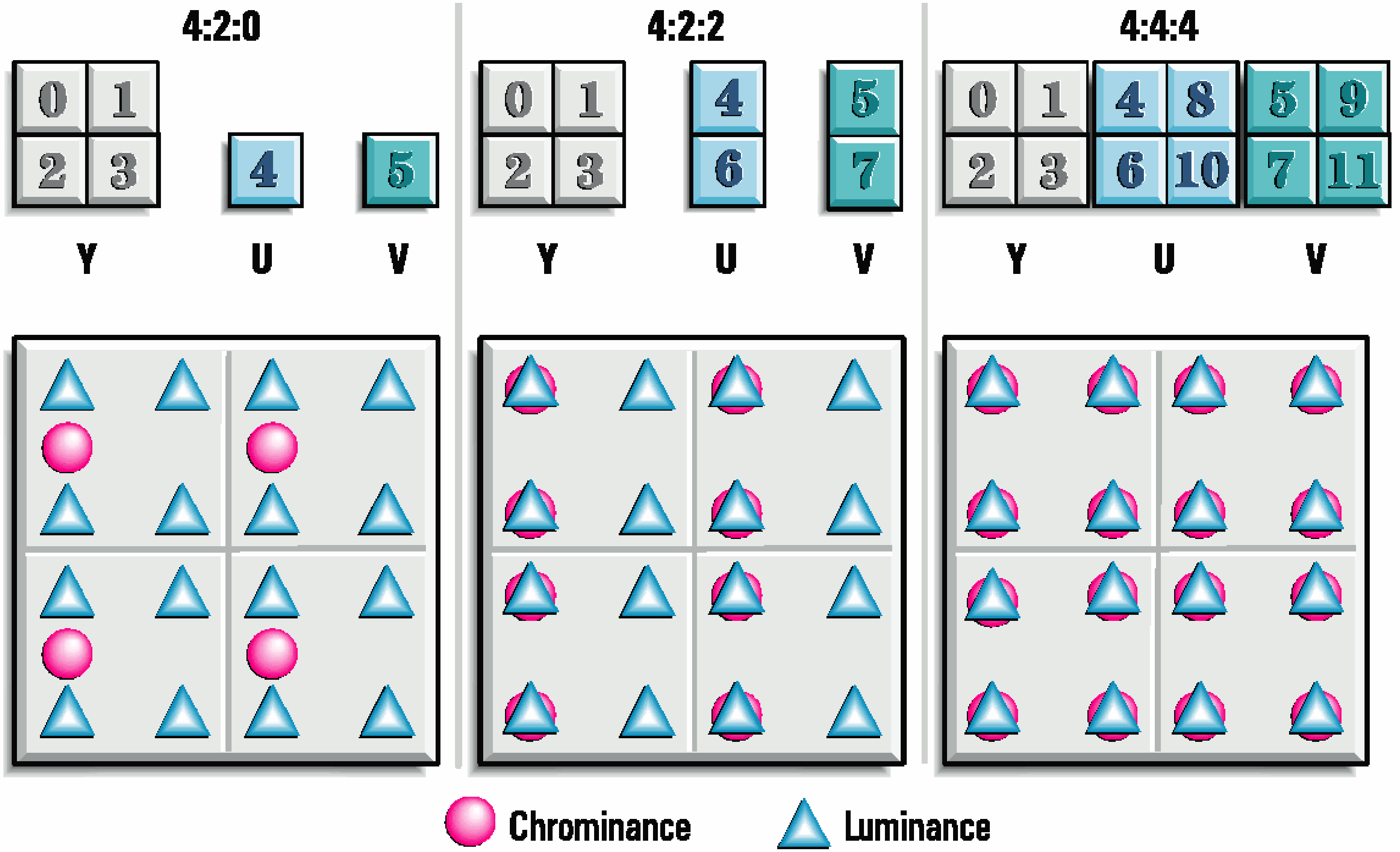
- RGB → YCrCb

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

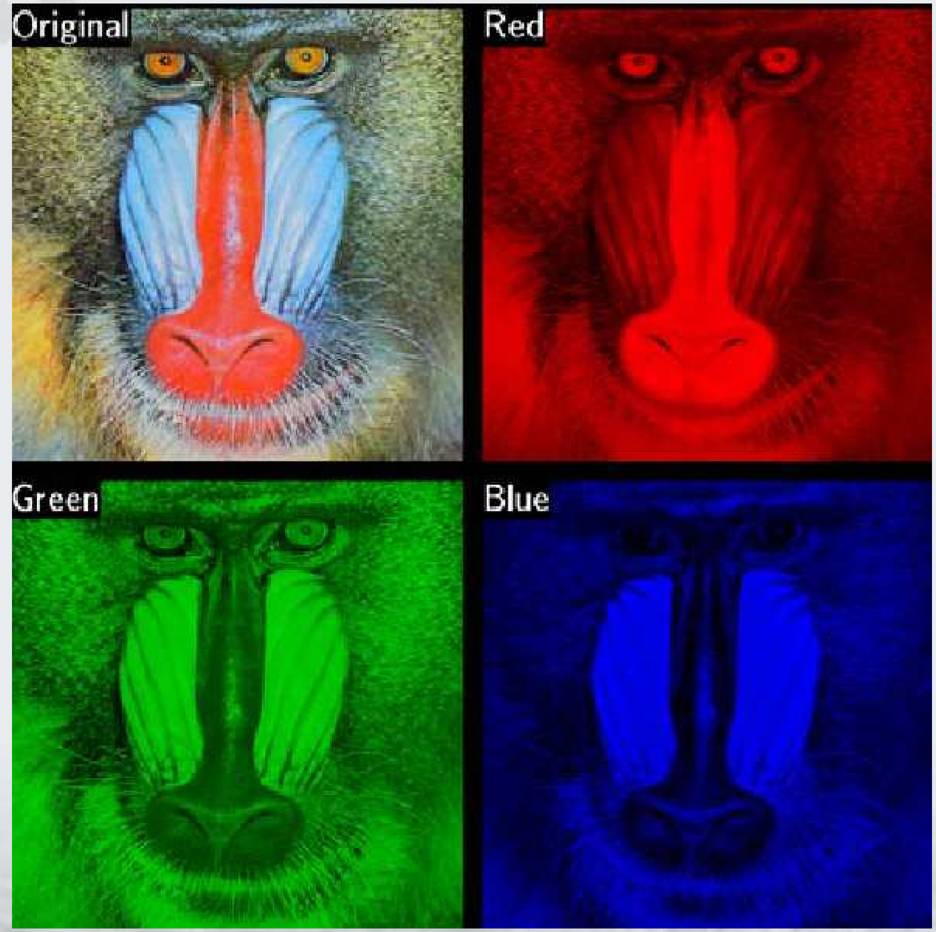
- Subsampling

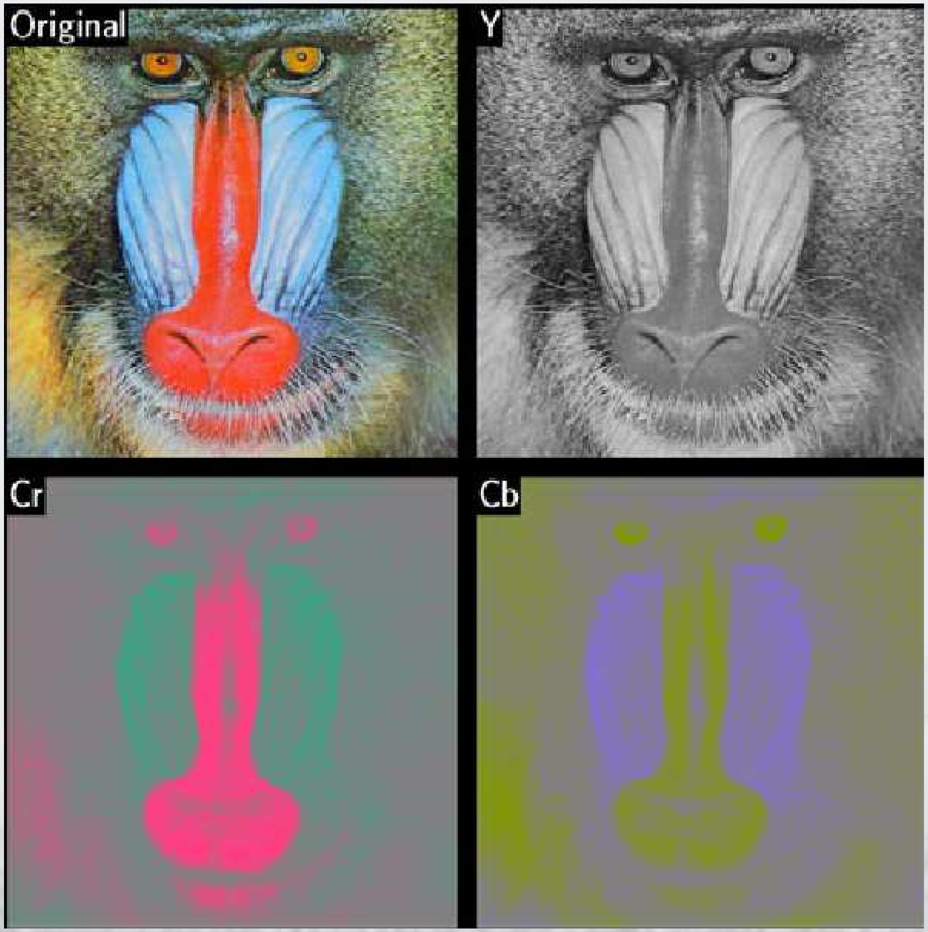
4:4:4 (no subsampling)
4:2:2 (Cr, Cb horizontal subsampling)
4:2:0 (Cr, Cb horizontal + vertical subsampling)

Chrominance subsampling



RGB vs YCC





Subsampling

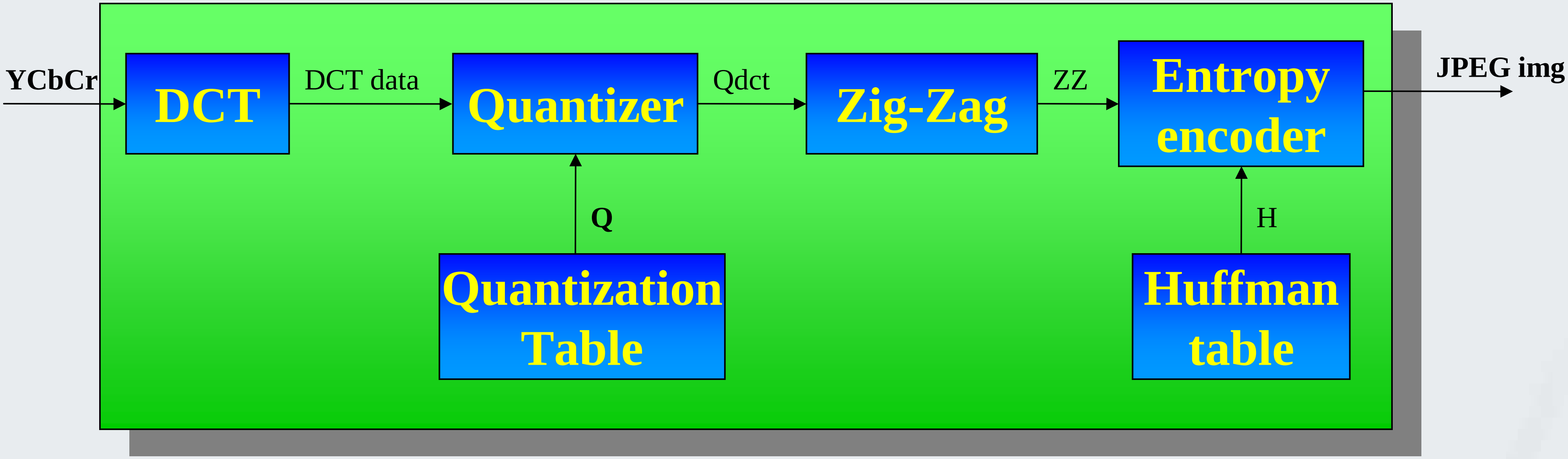
- Subsampling is allowed in all the components (Y,Cr,Cb)
- Only the chroma subsampling is usually used!!!

Original

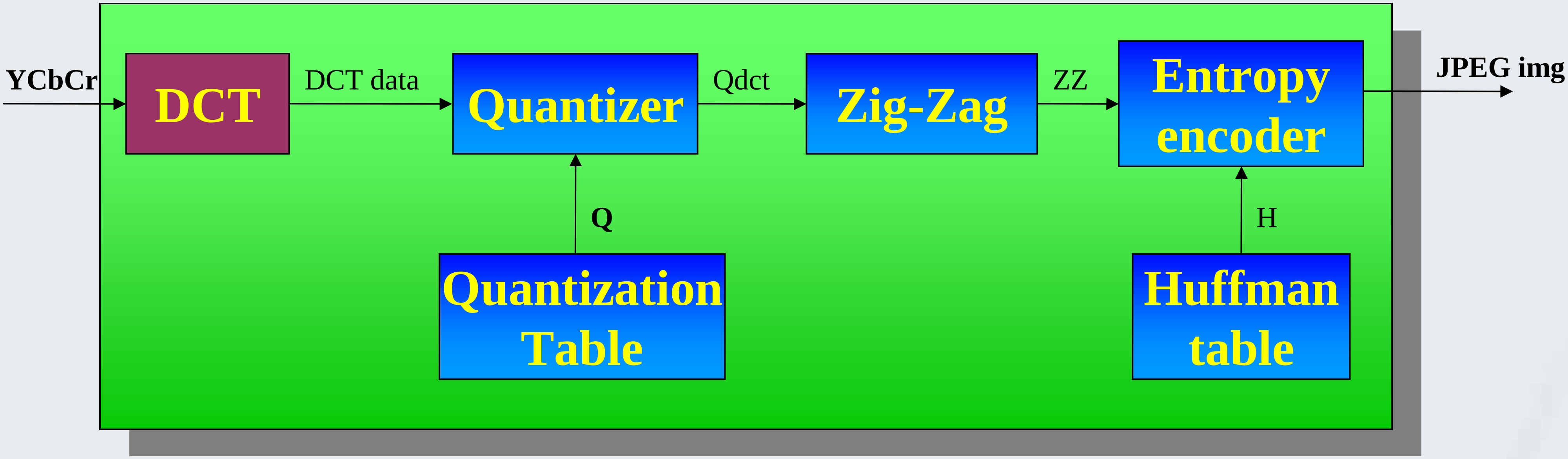
Luma Subsampled 8x

Chroma Subsampled 8x

JPEG


YCbCr
DCT
DCT data
Quantizer
Qdct
Zig-Zag
ZZ
Entropy encoder
JPEG img
Q
H
Quantization Table
Huffman table

JPEG


YCbCr
DCT
DCT data
Quantizer
Qdct
Zig-Zag
ZZ
Entropy encoder
JPEG img
Q
Quantization Table
H
Huffman table

Partition and DCT transform

Partition the input image into non-overlapping 8×8 blocks. The forward DCT is applied to each image block-by block by the forward JPEG DCT. Main advantages of DCT are:

- The energy compaction performance is nearly optimal closest to the KLT (Karhunen-Loeve Transform);
- The DCT coefficients are real numbers;
- DCT is a reversible linear transform and provides a set of orthonormal discrete basis functions;
- Many fast algorithms for forward and inverse DCT are known;

DCT formulas

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

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

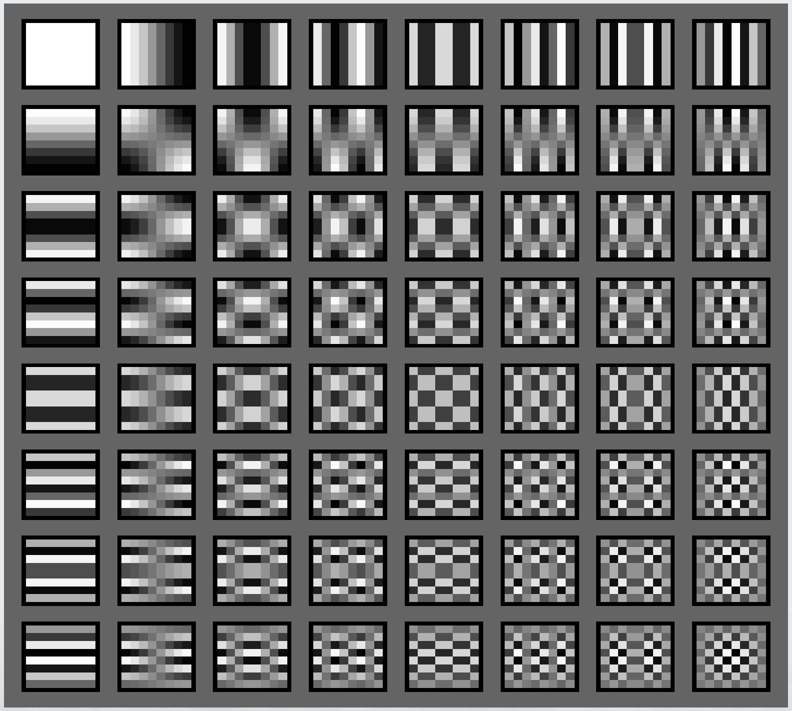
where:

$$C(u), C(v) = \frac{1}{\sqrt{2}} \text{ for } u, v = 0;$$

$$C(u), C(v) = 1 \quad \text{otherwise}$$

DCT basis

The 64 (8 x 8) DCT basis functions:



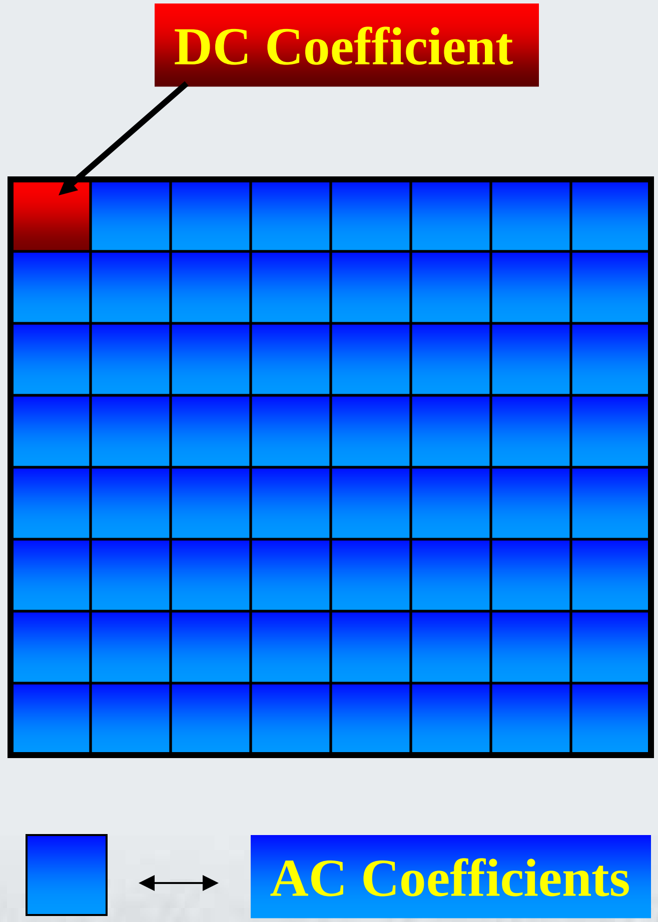
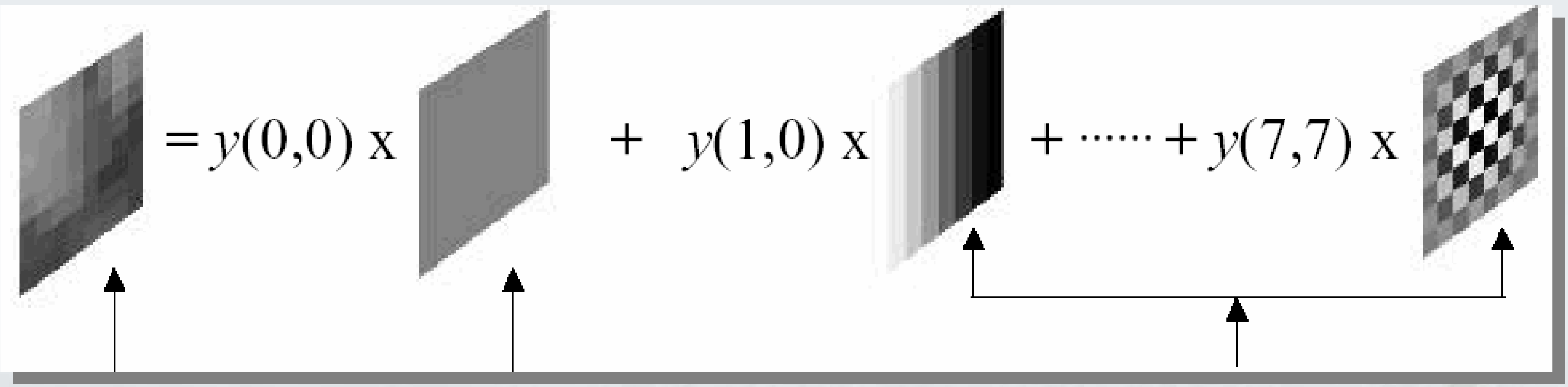


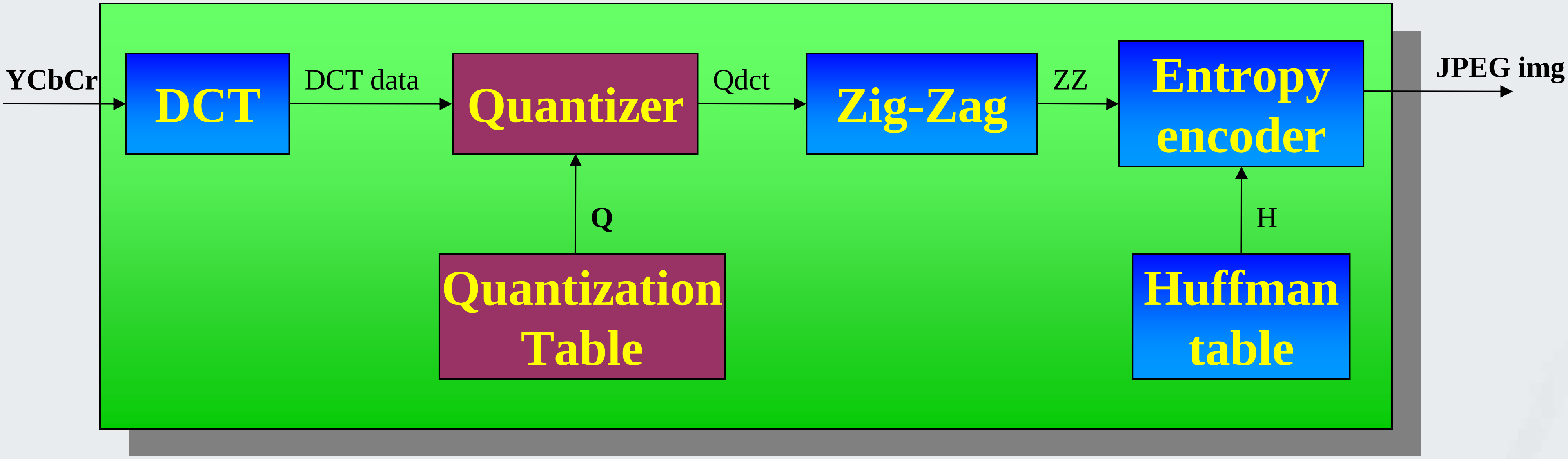
Image Representation with DCT

DCT coefficients can be viewed as weighting functions that, when applied to the 64 cosine basis functions of various spatial frequencies (8 x 8 templates), will reconstruct the original block.



Original image block   **DC** (flat) basis function   **AC** basis functions

JPEG



YCbCr
DCT
DCT data
Quantizer
Qdct
Zig-Zag
ZZ
Entropy encoder
JPEG img
Q
Quantization Table
H
Huffman table

DCT Coefficients Quantization

- **The DCT coefficients are quantized to limited number of possible levels.**
- **The Quantization is needed to reduce the number of bits per sample.**

Example:

101000 = 40 (6 bits precision) → Truncates to 4 bits = 1000 = 8 (4 bits precision).

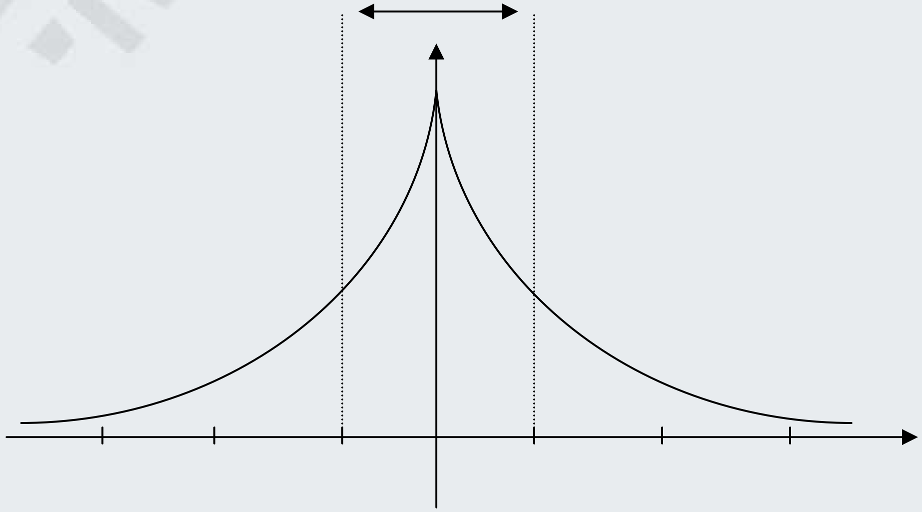
i.e. 40/5 = 8, there is a constant N=5, or the *quantization or quality factor* .

Formula:

$$F(u, v) = \text{round}[\, F(u, v) / Q(u, v)]$$

- **Q(u, v) = constant => Uniform Quantization.**
- **Q(u, v) = variable => Non-uniform Quantization.**

Quantization step



It is possible to approximate the statistical distribution of the AC DCT coefficients, both luminance and chrominance components, of a 8x8 block, by a Laplacian distribution in the following way:

$$p_i(x) = \lambda_i / 2 \; e^{-\lambda_i |x|} \qquad i = 1, 2, ..., 64;$$

where:

$\lambda_i = sqrt(2)/\sigma_i$;

σ_i = *i*-th DCT standard deviation;

EXAMPLE:

Q(u,v)= 8; Quantization Step

Round(256/8)= 32 Intervals;

[0, 8, 16, 24, 32, 40, ..., 256] - Reconstruction Levels

Standard Q-tables

Eye is most sensitive to low frequencies (upper left corner), less sensitive to high frequencies (lower right corner)

Luminance Quantization Table

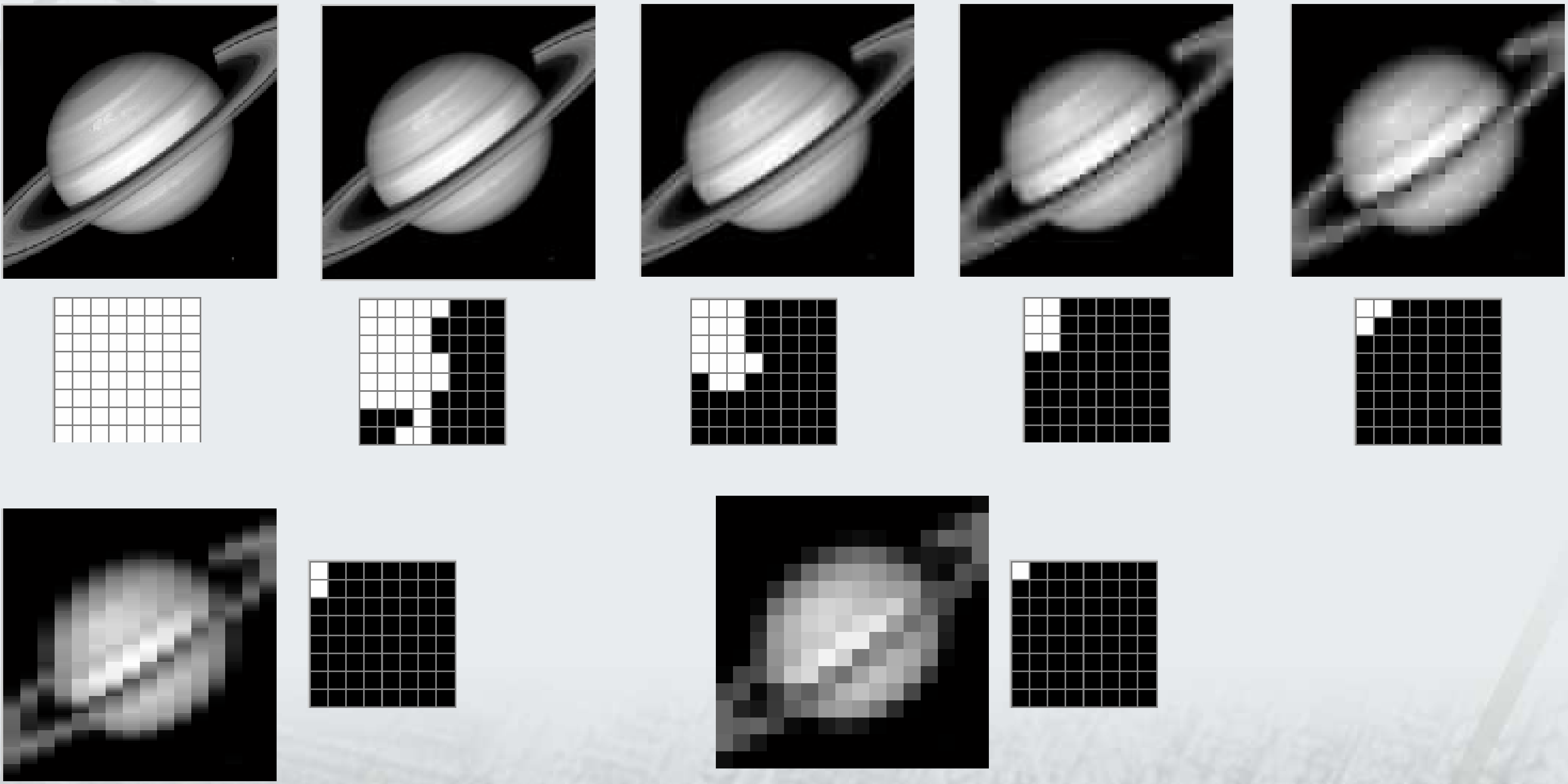
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

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 numbers in the above quantization tables can be scaled up (or down) to adjust the so called **Quality Factor QF.** (i.e. $Q^*(u,v) = QF \times Q(u,v)$)
Custom quantization tables can also be put in image/scan header.

Zonal Quantization example



An example

$$X = \begin{bmatrix} 168 & 161 & 161 & 150 & 154 & 168 & 164 & 154 \\ 171 & 154 & 161 & 150 & 157 & 171 & 150 & 164 \\ 171 & 168 & 147 & 164 & 164 & 161 & 143 & 154 \\ 164 & 171 & 154 & 161 & 157 & 157 & 147 & 132 \\ 161 & 161 & 157 & 154 & 143 & 161 & 154 & 132 \\ 164 & 161 & 161 & 154 & 150 & 157 & 154 & 140 \\ 161 & 168 & 157 & 154 & 161 & 140 & 140 & 132 \\ 154 & 161 & 157 & 150 & 140 & 132 & 136 & 128 \end{bmatrix}$$

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

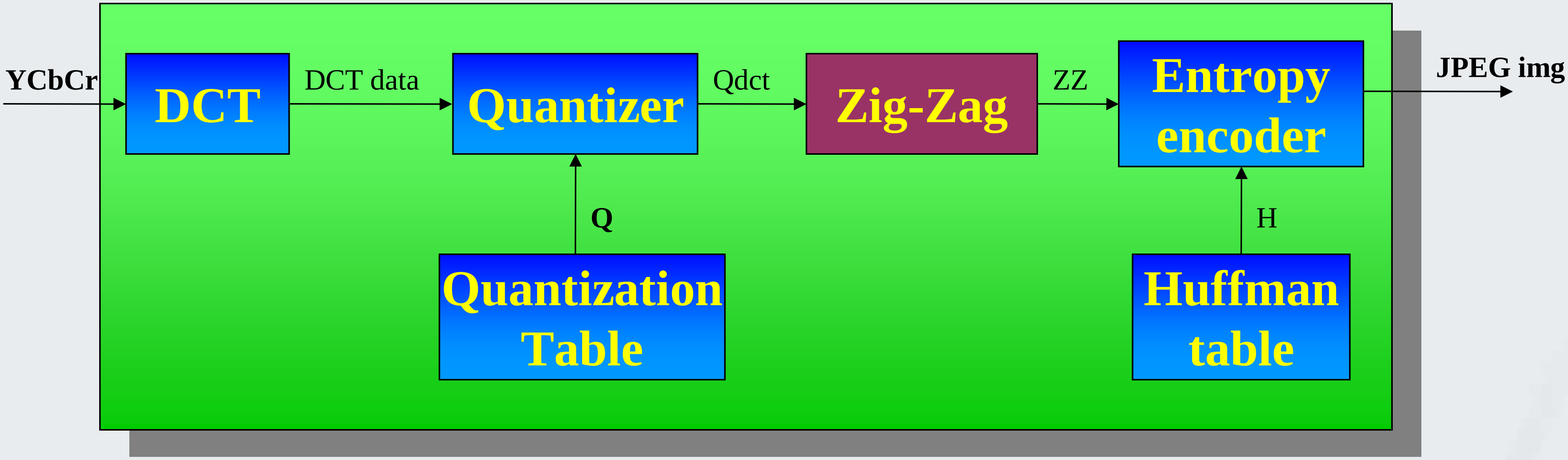
$$Y = \begin{bmatrix} 214 & 49 & -3 & 20 & -10 & -1 & 1 & -6 \\ 34 & -25 & 11 & 13 & 5 & -3 & 15 & -6 \\ -6 & -4 & 8 & -9 & 3 & -3 & 5 & 10 \\ 8 & -10 & 4 & 4 & -15 & 10 & 6 & 6 \\ -12 & 5 & -1 & -2 & -15 & 9 & -5 & -1 \\ 5 & 9 & -8 & 3 & 4 & -7 & -14 & 2 \\ 2 & -2 & 3 & -1 & 1 & 3 & -3 & -4 \\ -1 & 1 & 0 & 2 & 3 & -2 & -4 & -2 \end{bmatrix}$$

$$Z = \begin{bmatrix} 13 & 4 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & -2 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

X: original values; Y: DCT values; Q: Quantization table; Z: Quantized values

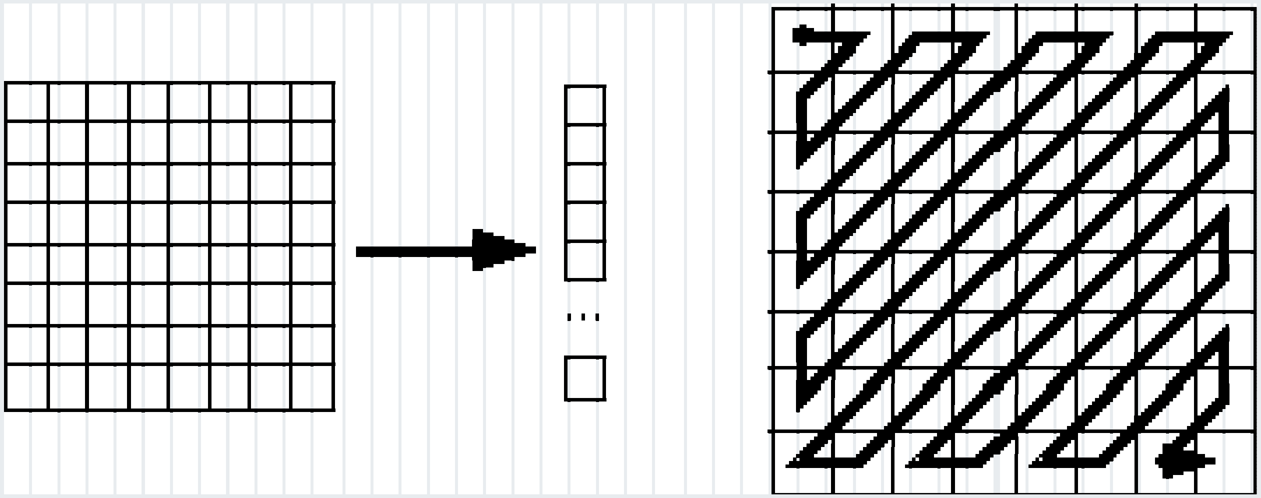
$$z_{ij} = round(y_{ij} / q_{ij})$$

JPEG



YCbCr
DCT
DCT data
Quantizer
Qdct
Zig-Zag
ZZ
Entropy encoder
JPEG img
Q
H
Quantization Table
Huffman table

Zig-Zag Ordering

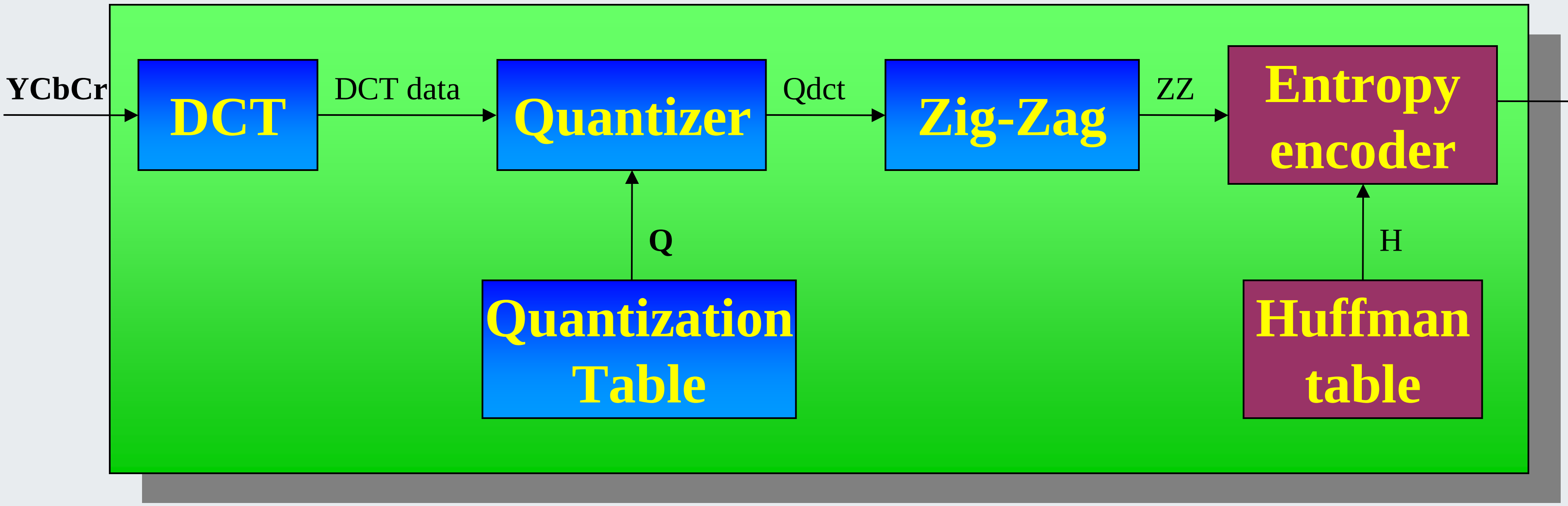


After Quantization, the DCT is separated into a DC coefficient and AC coefficients, which are reordered into a 1-D format (**8 x 8 to a 1 x 64 vector, in a suitable perceptive way)** using a zigzag pattern in order to create long runs of zero-valued coefficients.

The DC coefficient is directly correlated to the mean of the 8-by-8 block (upperleft corner). All DC coefficients are combined into a separate bit stream.

The AC coefficients are the values of the cosine basis functions (all other values).

JPEG


YCbCr
DCT
DCT data
Quantizer
Qdct
Zig-Zag
ZZ
Entropy encoder
JPEG img
Q
H
Quantization Table
Huffman table

Entropy encoder

- It is a Variable Length Coding algorithm
- It is based on Huffman algorithm
- Run length / variable length encoding
- Encode separately the DC and the AC coeffs
- 2 Huffman tables for the DC components
- 2 Huffman tables for the AC components
- All the Huffman tables can be modified (i.e. they can be adapted to the image content)

DC coefficients encoding

- **Encode the difference from the DC component of previous 8× 8 block, i.e.** ***lossless DPCM (Differential Pulse Code Modulation)*****, using the previous block DC as 1-D predictor.**

- **DC components are large and varied slowly, often close to previous value.**

AC Coefficients encoding

- **AC coefficients: using the zigzag ordering to create a 1-D sequence**
- **A list of events is built collecting run of zeros and the non-zero element**
- **The 1-D sequence is encoded in a collection of 2-tuples (*skip,value*), where *skip* is the number of zeroes and *value* is the next non-zero component.**

Entropy Coding

Categorize DC values into SIZE (number of bits needed to represent) and actual bits.

SIZE	Value
1	-1, 1
2	-3, -2, 2, 3
3	-7..-4, 4..7
4	-15..-8, 8..15
. . . . . .	
10	-1023..-512, 512..1023

Example: if DC value is 4, 3 bits are needed. Send off SIZE as Huffman symbol, followed by actual 3 bits.

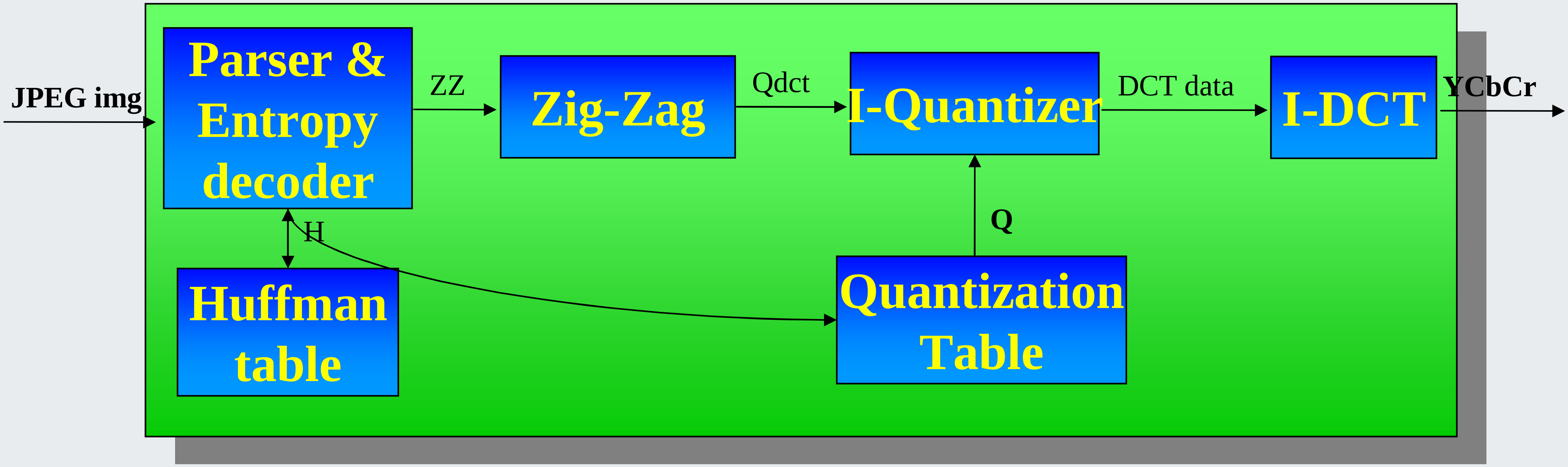
For AC components the non-zero *value* fixes the SIZE; then the Huffman code is composed by two parts:

Symbol_1: (skip, SIZE), [Huffman code]

Symbol_2: value. [SIZE bits are used]

Huffman Tables can be customizable (sent in header) or default.

JPEG decoding schema



JPEG decoding scheme

Entropy Decode and Zigzag Deordering

The entropy encoded data is first decoded and the data is de-zigzagged to recover the quantized values *FQ*(*u,v*) exactly.

Reconstruction of Quantized Coefficient Matrix

The *FQ* (*u,v*) is inverse scaled using *Q*(*u,v*) as
$$F(u,v)= Q(u,v)\ FQ(u,v)$$

Inverse DCT Transform (block by block)

$YCbCr \rightarrow RGB$

Four JPEG modes

- **Sequential/Baseline Mode;**
- **Lossless Mode;**
- **Progressive Mode;**
- **Hierarchical Mode;**

In *Motion JPEG*, Sequential JPEG is applied to each image in a video.

Examples (1/3)

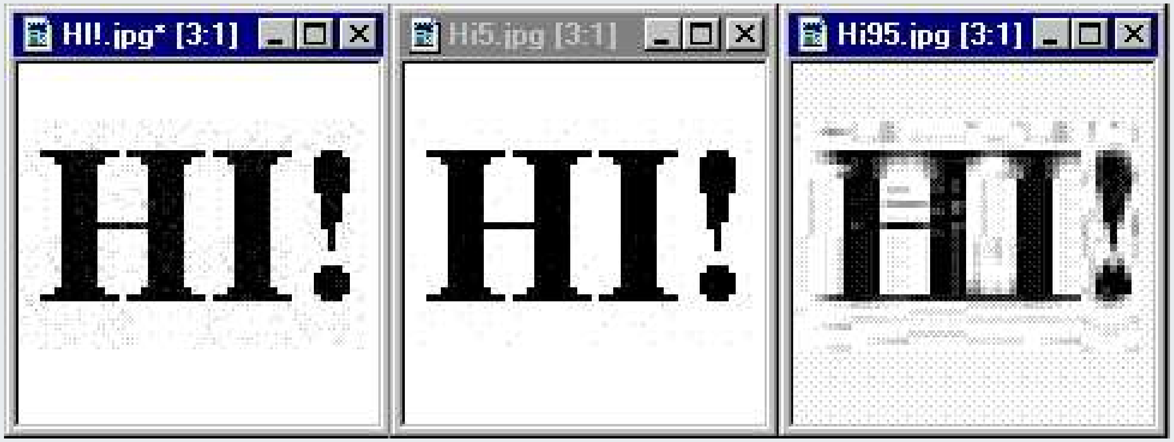


Original **CR. 75:1 QF=10** **CR. 110:1 QF=5**

Examples (2/3)

Original Uncompressed (3.2MB) **JPEG low level (179 KB)** **JPEG High level (15 KB)**

Examples (3/3)



JPEG DCT Pros and Cons

- **Advantages**

– Memory efficient, Low complexity, Compression efficiency, Visual model utilization, Robustness

- **Disadvantages**

– Single resolution, Single quality, No target bit rate, No lossless capability, No tiling, No region of interest, Blocking artifacts, Poor error resilience