International Standard of Still Color Image ITU/ISO JPEG

- History Introduction
- DCT-based Coding
- Color Images
- Quantization
- Entropy coding
- JPEG Modes of Operation
- Implementation Issues
- Extensions

Historical Introduction to JPEG Standard

- Since 1986, joint meeting for *International Telecommunication Union (ITU)* and the *International Organization for Standardization (ISO)* to establish the standard for ***multilevel color still images.*** In 1987, the *International Electrotechnical Committee (IEC)* joined in.
- Known as JPEG, Joint Photographic Experts Group
- Schedules:
  – 1988, select the DCT-based method
  – 1988-1990, simulating, testing and documenting
  – 1991, draft
  – 1992, international standard
- Official document:
  – ISO/IEC international standard 10918-1(set requirements and guides) (-2, for compliance tests, -3, for extensions)
    – digital compression and coding of continuous-tone still images
  – ITU-T Recommendation T.81

Goals of JPEG

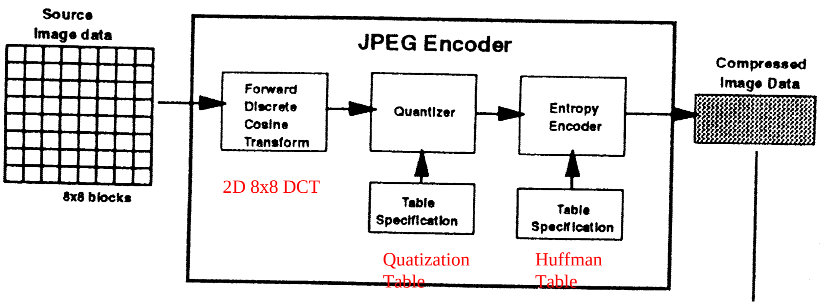
- Be as close as possible to the state of the art in image compression
- Allow applications to tradeoff easily between desired compression and image quality
- Work independently of image type
- Have modest computational complexity that would allow software-only implementation even in low-end computers
- Allow both sequential(single scan) and progressive(multiple scan) coding
- Offers the opinion for hierarchical encoding

3

@NTUEE DSP/IC Lab

DCT-Based Coding

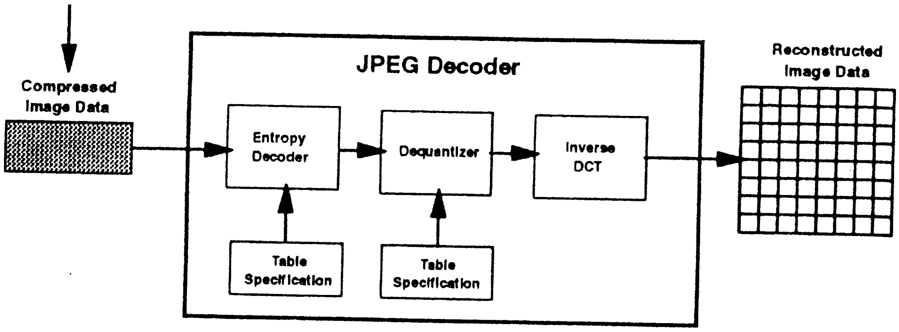
JPEG baseline system



4

@NTUEE DSP/IC Lab

DCT-Based Coding



@NTUEE DSP/IC Lab

Zig-Zag Scan

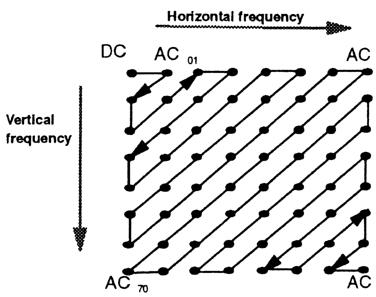


Figure 5.2 Zig-zag ordering of AC coefficients.

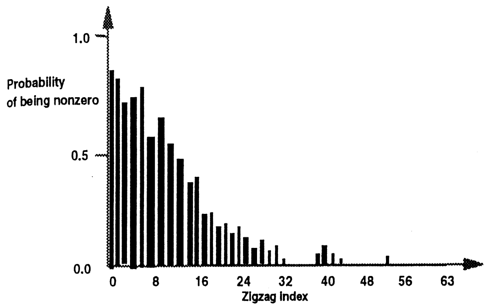


Figure 5.3 The probability of being nonzero of zig-zag ordered AC coefficients.

@NTUEE DSP/IC Lab

Design for Quantization Table

- Bit-rate control technique:
  - to allocate more bits to the coefficients with large variances.

8	6	5	8	12	20	26	30
6	6	7	10	13	29	30	28
7	7	8	12	20	29	35	28
7	9	11	15	26	44	40	31
9	11	19	28	34	55	52	39
12	18	28	32	41	52	57	46
25	32	39	44	52	61	60	51
36	46	48	49	56	50	52	50

9	9	12	24	50	50	50	50
9	11	13	33	50	50	50	50
12	13	28	50	50	50	50	50
24	33	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

16	17	18	19	20	21	22	23
17	18	19	20	21	22	23	24
18	19	20	21	22	23	24	25
19	20	21	22	23	24	25	26
20	21	22	23	24	25	26	27
21	22	23	24	25	26	27	28
22	23	24	25	26	27	28	29
23	24	25	26	27	28	29	30

16	16	19	22	26	27	29	34
16	16	22	24	27	29	34	37
19	22	26	27	29	34	34	38
22	22	26	27	29	34	37	40
22	26	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	27	29	34	38	46	56	69
27	29	35	38	46	56	69	83

Table 5.1 Four quantization tables for compliance testing of generic JPEG encoders and decoders.

Quantization Table

- For luminance

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

- For chrominance

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	59	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

@NTUEE DSP/IC Lab

Predictive Coding of DC Coefficients

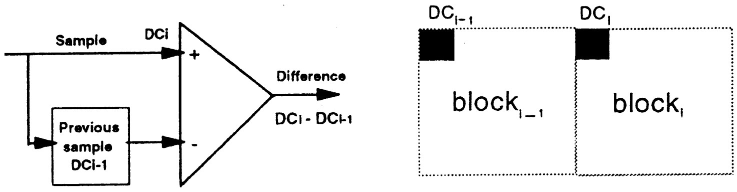


Figure 5.4 Predictive coding of DC coefficients. The difference between the present and the previous DC coefficients is calculated and then coded using JPEG.

Entropy Coding

- Huffman coding and Arithmetic coding
- Huffman encoder (two steps)
  - forming *intermediate symbol sequence*
  - converting *intermediate symbol sequence* into *binary sequence* using Huffman table
- DC : (Size, amplitude),
  - where *size* defines the number of bits required to represent the amplitude, and *amplitude* is the 1's complement amplitude of differential.
  - Only the *size* is huffman coded.
- AC : (runlength, size, amplitude)

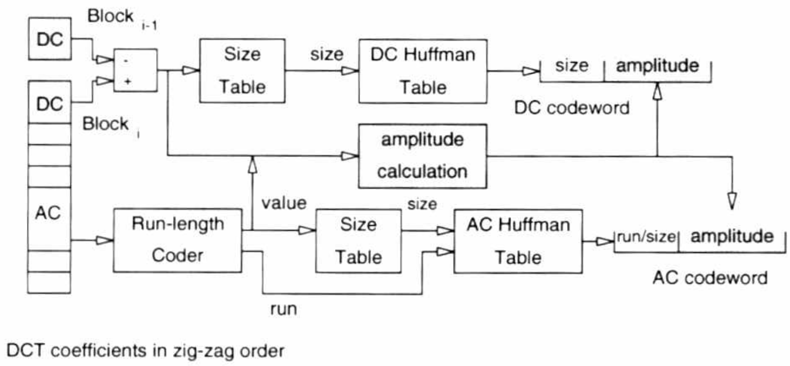


Figure 5.5 Huffman coding in baseline JPEG.

11

@NTUEE DSP/IC Lab

Huffman table Example

Size	Amplitude range
1	(-1,1)
2	(-3,-2) (2,3)
3	(-7..-4) (4..7)
4	(-15..-8) (8..15)
5	(-31..-16) (16..31)
6	(-63..-32) (32..63)
7	(-127..-64) (64..127)
8	(-255..-128) (128..255)
9	(-511..-256) (256..511)
10	(-1023..-512) (512..1023)

Table 5.2 Huffman coding of Symbols-2.

12

@NTUEE DSP/IC Lab

DC-Table

- For luminance

Size	length	code word
0	2	0 0
1	3	0 1 0
2	3	0 1 1
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

- For chrominance

Size	length	code word
0	2	0 0
1	2	0 1
2	2	1 0
3	3	110
4	4	1110
5	5	11110
6	6	111110
7	7	1111110
8	8	11111110
9	9	111111110
10	10	1111111110
11	11	11111111110

Value	Run/Size	Huffman Code	Amplitude	Total Bits
16	0/5	11010	10000	10
-21	0/5	11010	01010	10
10	0/4	1011	1010	8
-15	0/4	1011	0000	8
3	3/2	111110111	11	11
-2	0/2	01	01	4
2	1/2	11011	10	7
-3	0/2	01	00	4
2	5/2	11111110111	10	13
-1	0/1	00	0	3
EOB	0/0	1010		4

Table 5.1 Example for the Huffman coding of AC coefficients.

Example for AC

- The sequences are: 0,0,0,0,0,0,-18
  - ---> (6, 5, 01101) means *run* is 6, size 5, and amplitude 01101 (-18)
  - huffman coded for (6/5) is 1101, then the codeword for -18 is 110101101
- Special Cases:
  - 1) The run length value may be larger than 15. In that case, JPEG uses the symbol (15/0) to denote a run-length of 15 zeros followed by a zero. Such symbol can be cascaded as needed; however, the codeword for the last AC must have a nonzero amplitude.
  - 2) If after a nonzero AC value all the remaining coefficients are zero, then the special symbol (0/0) denotes an end-of-block(EOB).

AC-Table

- For luminance

Run/Size	length	code word
0/0	4	1010 (EOB)
0/1	2	0 0
0/2	2	0 1
0/3	3	100
0/4	4	1011
0/5	5	11010
0/6	7	1111000
…	…	…
F/0	11	11111111001
…	…	…
F/A	16	1111111111111110

- For chrominance

Run/Size	length	code word
0/0	2	0 0 (EOB)
0/1	2	0 1
0/2	3	100
0/3	4	1010
0/4	5	11000
0/5	5	11001
0/6	6	111000
…	…	…
F/0	10	1111111010
…	…	…
F/A	16	1111111111111110

@NTUEE
DSP/IC Lab

Example

(RUNLENGTH, SIZE)	Code Word
(0,0) EOB	1010
(0,1)	00
(0,2)	01
(0,3)	100
(1,2)	11011
(2,1)	11100
(3,1)	111010
(4,1)	111011
(5,2)	11111110111
(6,1)	1111011
(7,1)	11111010

(a) Original 8x8 block

140	144	147	140	140	155	179	175
144	152	140	147	140	148	167	179
152	155	136	167	163	162	152	172
168	145	156	160	152	155	136	160
162	148	156	148	140	136	147	162
147	167	140	155	155	140	136	162
136	156	123	167	162	144	140	147
148	155	136	155	152	147	147	136

(b) Shifted block

12	16	19	12	11	27	51	47
16	24	12	19	12	20	39	51
24	27	8	39	35	34	24	44
40	17	28	32	24	27	8	32
34	20	28	20	12	8	19	34
19	39	12	27	27	12	8	34
8	28	-5	39	34	16	12	19
20	27	8	27	24	19	19	8

(c) Block after FDCT Eq. (5.1)

185	-17	14	-8	23	-9	-13	-18
20	-34	26	-9	-10	10	13	6
-10	-23	-1	6	-18	3	-20	0
-8	-5	14	-14	-8	-2	-3	8
-3	9	7	1	-11	17	18	15
3	-2	-18	8	8	-3	0	-6
8	0	-2	3	-1	-7	-1	-1
0	-7	-2	1	1	4	-6	0

(d) Quantization Table (quality=2)

3	5	7	9	11	13	15	17
5	7	9	11	13	15	17	19
7	9	11	13	15	17	19	21
9	11	13	15	17	19	21	23
11	13	15	17	19	21	23	25
13	15	17	19	21	23	25	27
15	17	19	21	23	25	27	29
17	19	21	23	25	27	29	31

(e) Block after quantization Eq. (5.2)

61	-3	2	0	2	0	0	-1
4	-4	2	0	0	0	0	0
-1	-2	0	0	-1	0	-1	0
0	0	1	0	0	0	0	0
0	0	0	0	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

(f) Zig-zag sequence

61,-3,4,-1,-4,2,0,2,-2,0,0,0,0,0,2,0,0,0,1,0,0,0,0,0,0,-1,0,0,-1,0,0,
0,0,-1,0,0,0,0,0,0,0,-1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

(g) Intermediate symbol sequence

(6)(61),(0,2)(-3),(0,3)(4),(0,1)(-1),(0,3)(-4),(0,2)(2),(1,2)(2),(0,2)(-2),
(0,2)(-2),(5,2)(2),(3,1)(1),(6,1)(-1),(2,1)(-1),(4,1)(-1),(7,1)(-1),(0,0)

(e) Encoded bit sequence (total 98 bits)

(110)(111101) (01)(00) (100)(100) (00)(0) (100)(001) (01)(10)
(11011)(10) (01)(01) (01)(01) (11111110111)(10) (111010)(1)
(1111011)(0) (11100)(0) (111011)(0) (11111010)(0) (1010)

Figure 5.5 Step-by-step procedure in JPEG sequential encoding of a 8x8 block.

Encoding Grayscale Images

Quality factor	Original number of bits	Compressed number of bits	Compression ratio (Cr)	Bits/pixel (Nb)	RMS error	Execution times [ms] SUN SPARC 10/41	Execution times [ms] SUN IPC
1	512,000	48,021	10.66	0.75	2.25	0.59	6.31
2	512,000	30,490	16.79	0.48	2.75	0.59	6.22
4	512,000	20,264	25.27	0.32	3.43	0.58	6.39
8	512.000	14,162	36.14	0.22	4.24	0.59	6.44
15	512,000	10,479	48.85	0.16	5.36	0.58	6.45
25	512,000	9,034	56.64	0.14	6.40	0.58	6.32
DC only	512,000	7,688	66.60	0.12	7.92	0.57	6.25

Table 5.5 Results of JPEG compression of grayscale image 'Lisa' (320x240 pixels).

JPEG Compression of Color Images

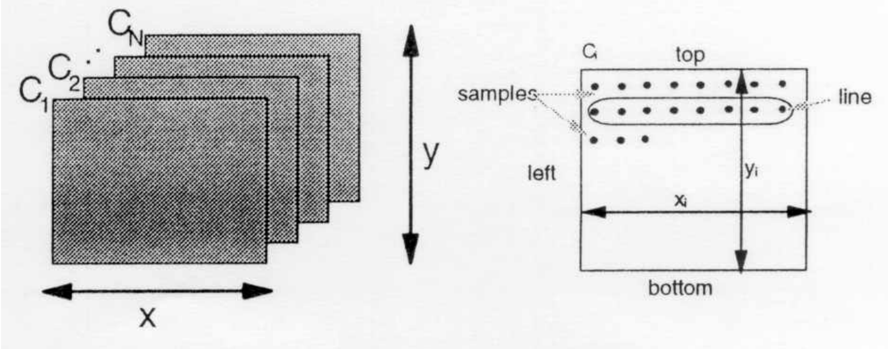


Figure 5.7 JPEG color image model.

19

Resolutions of Color Images

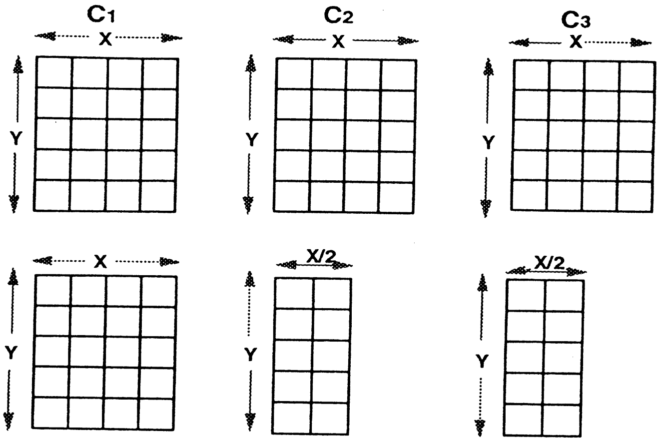


Figure 5.8 A color image with 3 components: (a) with same resolutions, (b) with different resolutions.

20

Non-interleaved & Interleaved Data Ordering

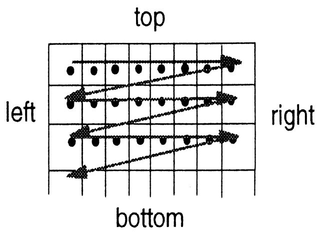
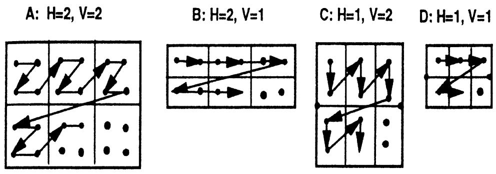


Figure 5.9 Noninterleaved data ordering in JPEG compression of multipl components images.

The RGB or YUV image are stored as three separate frames.



Data units	A	B	C	D
MCU1 =	{A00,A01,A10,A11,	B00,B01	C00,C10,	D00}
MCU2 =	{A02,A03,A12,A13,	B02,B03,	C01,C11,	D01}
MCU3 =	{A04,A05,A14,A15,	B04,B05	C02,C12,	D02}
MCU4 =	{A20,A21,A30,A31,	B10,B11	C20,C30,	D10}

Figure 5.10 Interleaved data ordering in JPEG compression of color images. The MCU consists of data units taken from each component A, B, C, and D.

MCU: minimum coded unit

Experimental Profiling Data

OPERATION	JPEG COMPRESSION		JPEG DECOMPRESSION	
	Time [sec]	Percentage	Time [sec]	Percentage
Read/Write File	0.75	11	0.34	6
RGB/YUV Conversion	1.40	21	1.14	19
Reorder from/for YUV	0.28	4	0.49	8
FDCT/IDCT	2.87	43	2.90	48
Quantization/ Dequantization	0.47	7	0.37	6
Huffman enc./dec.	0.87	13	0.70	12
Write/Read File I/O	0.01	1	0.06	1
TOTAL TIME	**6.65**	**100%**	**6.00**	**100%**

Table 5.6 Experimental data: JPEG compression of a color image.

Operation Models of JPEG

- Sequential DCT-based encoding
  - image blocks are coded in a scan-like sequence, from left to right and from top to bottom
- Progressive DCT-based encoding
  - the coding is completed in multiple scan, the first scan yields the full image but without all the details, which are provided in successive scans.
  - Two procedures:
    - Spectral Selection (decoder side only)
    - Successive Approximation (codec)
- Lossless encoding
  - predictive coding only
- Hierarchical encoding

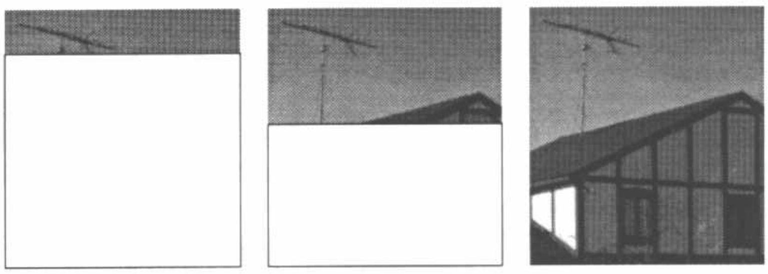


Figure 5.8 Example of sequential coding.

@NTUEE
DSP/IC Lab

Progressive JPEG Compression

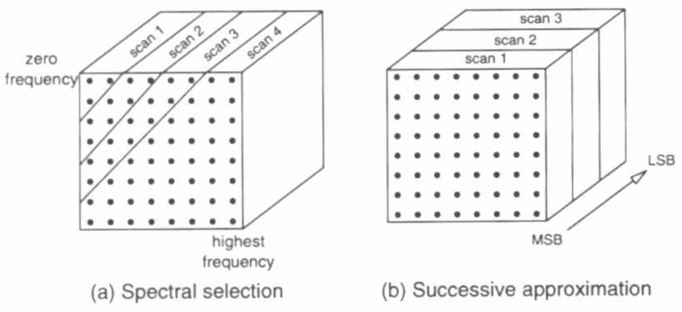


Figure 5.9 Description of progressive coding in JPEG.

@NTUEE DSP/IC Lab

Progressive JPEG Compression

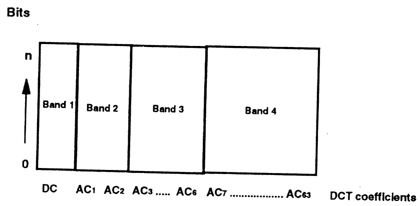


Figure 5.12 An example of applying the spectral selection progressive JPEG algorithm using four scans.

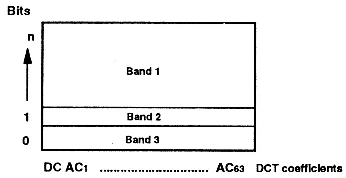


Figure 5.13 An example of applying the successive approximation progressive JPEG algorithm using three scans.

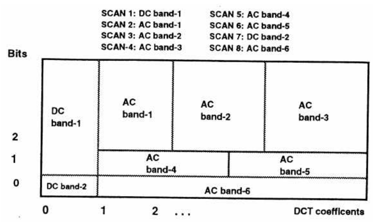


Figure 5.14 An example of applying combined progressive JPEG algorithm consisting of eight scans.

@NTUEE DSP/IC Lab

(a) DC only. (b) DC plus first three AC components. (c) Full resolution

Figure 5.10 Example of JPEG progressive coding using spectral selection. (a) Decoding of the DC coefficient only; (b) decoding of the DC and the first three (in zig-zag order) AC coefficients; (c) decoded image at full spectral resolution.

Experimental Data

	Spectral Seelection	Successive Approximation
SCAN 1	DC, AC1, AC2	All DCT - divided by 8
SCAN 2	AC3 - AC9	All DCT - divided by 4
SCAN 3	AC10 - AC35	All DCT divided by 2
SCAN 4	AC 36 - AC 63	All DCT - full resolution

Table 5.7 Four scans applied in the progressive JPEG experiments.

Scan number	Bits transmitted	Compression ratio	Bits/pixel	RMS error
1	29,005	17.65	0.45	19.97
2	37,237	7.73	1.04	13.67
3	71,259	3.72	2.15	7.90
4	92,489	3.01	2.66	4.59
Sequential JPEG	172,117	2.97	2.69	4.59

Table 5.8 Progressive spectral selection JPEG. Image 'Cheetah': 320x240 pixels = 512,000 bits.

Scan number	Bits transmitted	Compression ratio	Bits/pixel	RMS error
1	26,215	19.53	0.41	22.48
2	34,506	8.43	0.95	12.75
3	63,792	4.11	1.95	7.56
4	95,267	2.33	2.43	4.59
Sequential JPEG	172,117	2.97	2.69	4.59

Table 5.9 Progressive successive approximation JPEG. Image 'Cheetah': 320x240 pixels = 512,000 bits.

Sequential Lossless JPEG Compression

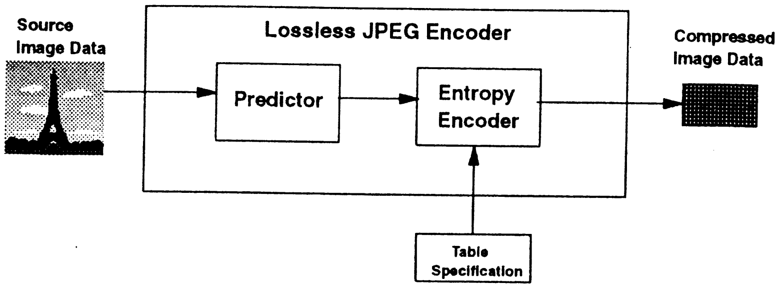


Figure 5.17 Block diagram of the lossless JPEG encoder.

Sequential Lossless JPEG Compression

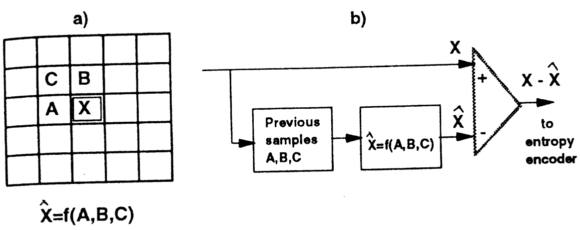
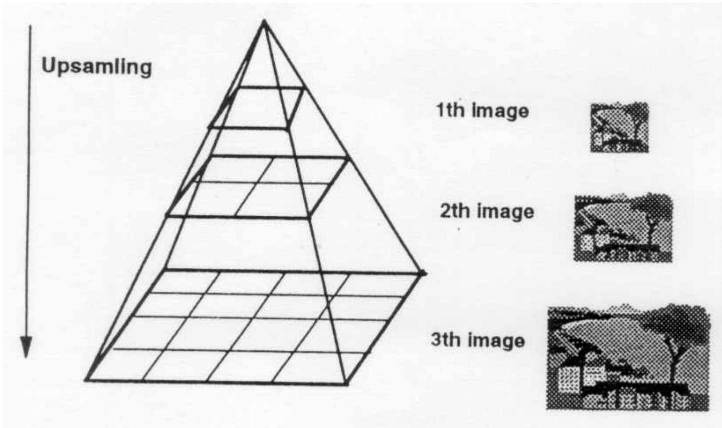


Figure 5.18 Lossless JPEG encoding: (a) Locations of four samples in the predictor, (b) Predictor block diagram.

Selection Value	Predictor Formula
0	no prediction
1	X = A
2	X = B
3	X = C
4	X = A+B-C
5	X = A + (B-C)/2
6	X = B+ (A-C)/2
7	X = (A+B)/2

Hierarchical JPEG Compression



31

@NTUEE DSP/IC Lab

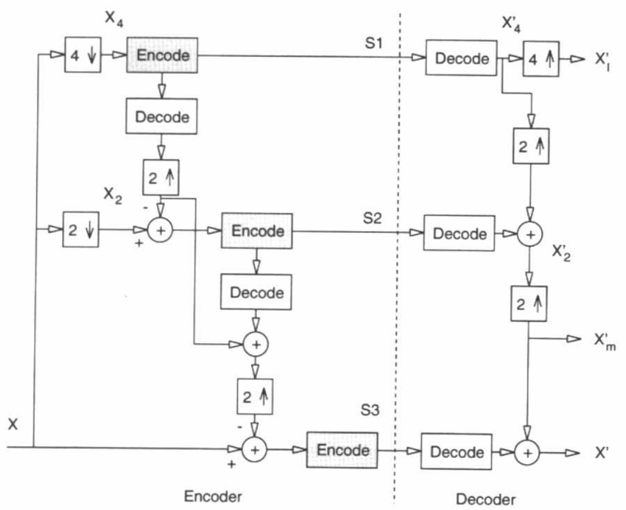


Figure 5.11 Three-level hierarchical coder.

32

@NTUEE DSP/IC Lab

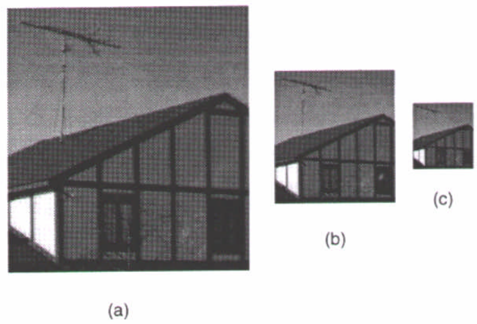
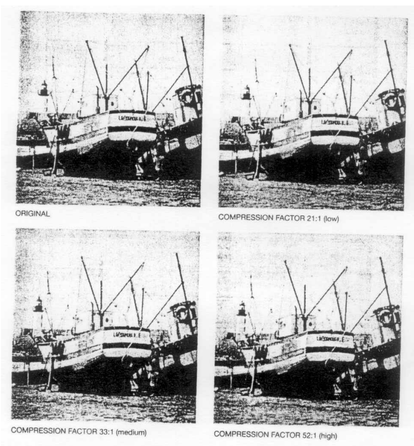


Figure 5.12 Example of hierarchical coding. (a) Original; (b) subsampled by a factor of two; (c) subsampled by a factor of four.

@NTUEE DSP/IC Lab

33

Results of Hierarchical Coding



@NTUEE DSP/IC Lab

34

Implementation Issues

- Results of JPEG Baseline Algorithm
- Hardware v.s. Software implementation
- DCT/IDCT complexity
- Arithmetic precision requirements
  - dynamic range
- JPEG Coding tables
- Color conversion
- JPEG file interchange format

CR=2, MSE=2.71, PSNR=49.86

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 129K

CR=3, MSE=13.34, PSNR=42.93

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 80K

37

@NTUEE
DSP/IC Lab

CR=4, MSE=19.91, PSNR=41.19

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 62K

38

@NTUEE
DSP/IC Lab

CR=8, MSE=43.10, PSNR=37.84

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 32K

@NTUEE DSP/IC Lab

39

CR=10, MSE=55.22, PSNR=36.76

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 25.2K

@NTUEE DSP/IC Lab

40

CR=20, MSE=114.64, PSNR=33.59

Original Lena Image
File Size = 256K

JPEG Compressed Image
File Size = 13K

Subjective View of Lena Image

Original Image

CR=2

CR=3

CR=4

CR=6

CR=8

CR=10

CR=20

CR=3, MSE=159.01, PSNR=37.63

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 293K

@NTUEE DSP/IC Lab

43

CR=4, MSE=176.01, PSNR=37.18

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 217K

@NTUEE DSP/IC Lab

44

CR=8, MSE=308.99, PSNR=34.75

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 112K

45

@NTUEE DSP/IC Lab

CR=10, MSE=398.41, PSNR=33.64

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 90K

46

@NTUEE DSP/IC Lab

CR=16, MSE=566.96, PSNR=32.11

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 57K

CR=20, MSE=669.73, PSNR=31.39

Original Color Titanic Image (RGB) File Size = 900K

JPEG Compressed Titanic Image File Size = 45K

Subjective View of Titanic Image

JPEG Extensions and Applications

- Variable quantization
  - introduces a quantizer scale factor that may be used to scale the original quantization matrix on a block-by-block basis
  - allows for greater control on image quality and compression ratios
  - good for text and graphics combined images
- Selective refinement
  - allows the encoder to specify which area of the image are coded in greater detail than the rest of the image
- Tiling
  - allows an image to be subdivided into subimages, tiles
  - simple tiling and pyramid tiling
- Color Fascimile standard

Subjective View

YUV Converted

PSNR: 36.25
File length: 162k

RGB Directly

PSNR: 44.63
File length: 154+143+146k

Subjective View(Con’t)

YUV Converted

PSNR: 35.57
File length: 133k

RGB Directly

PSNR: 41.49
File length: 131+119+123k

Subjective View(Con't)

YUV Converted

PSNR: 34.97
File length: 112k

RGB Directly

PSNR: 39.33
File length: 112+101+106k

@NTUEE DSP/IC Lab

53

Subjective View(Con't)

YUV Converted

PSNR: 33.22
File length: 76k

RGB Directly

PSNR: 35.35
File length: 76+68+72k

@NTUEE DSP/IC Lab

54

Comparisons

YUV			RGB		
File Length	CF	PSNR	File Length	CF	PSNR
162	4	36.25	131+119+123	6	41.49
133	6	35.57	112+101+106	8	39.33
112	8	34.97	76+68+72	16	35.35
76	16	33.22	154+143+146	4	44.63