The size of typical still image (1200x1600)

$$1200 \times 1600 \times 3 byte = 5760000 byte$$

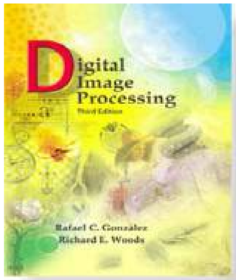
$$= 5{,}760 Kbyte = 5.76 Mbyte$$

The size of two hours standard television (720x480) movies

$$30 \frac{frame}{\sec} \times (760 \times 480) \frac{pixels}{frame} \times 3 \frac{bytes}{pixel} = 31{,}104{,}000 bytes / \sec$$

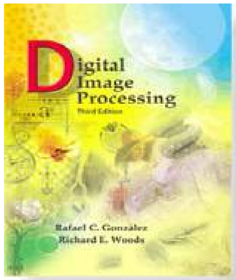
$$31{,}104{,}000 \times \frac{bytes}{\sec} \times (60 \times 60) \frac{\sec}{hour} \times 2 hours = 2.24 \times 10^{11} bytes$$

$$= 224 GByte.$$

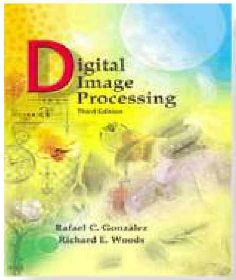


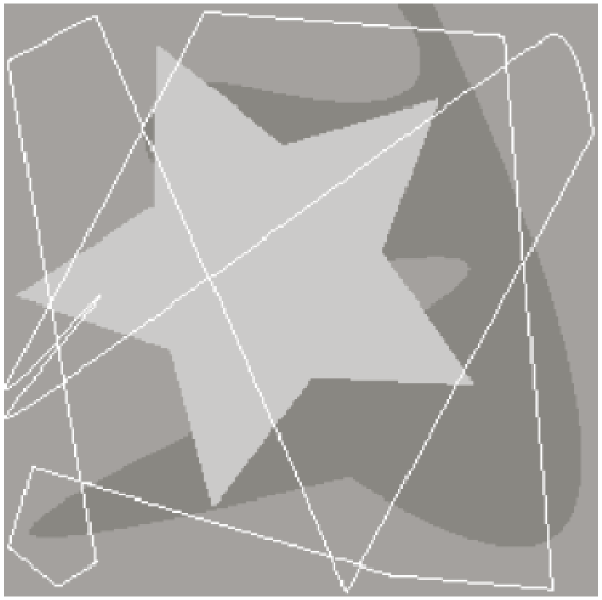
Data, Information, and Redundancy

- **Information**
- **Data** is used to represent information
- **Redundancy** in data representation of an information provides no relevant information or repeats a stated information
- Let n1, and n2 are data represents the same information. Then, the relative data redundancy R of the n1 is defined as
  $R = 1 - 1/C$ where $C = n1/n2$



- Redundancy in Digital Images
  - Coding redundancy
    usually appear as results of the uniform representation of each pixel
  - Spatial/Temopral redundancy
    because the adjacent pixels tend to have similarity in practical.
  - Irrelevant Information
    Image contain information which are ignored by the human visual system.

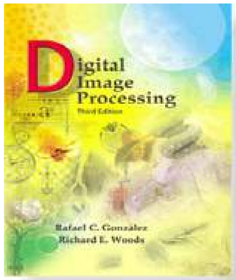




Coding Redundancy

Spatial Redundancy

Irrelevant Information

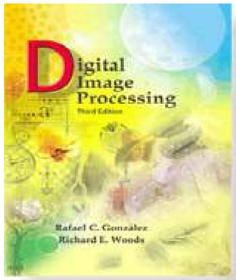


Coding Redundancy

- Assume the discrete random variable for r_k in the interval [0,1] that represent the gray levels. Each r_k occurs with probability p_k
- If the number of bits used to represent each value of r_k by $l(r_k)$ then

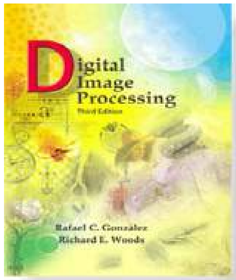
$$L_{avg} = \sum_{k=0}^{L-1} l(r_k) p(r_k)$$

- The average code bits assigned to the gray level values.
- The length of the code should be inverse proportional to its probability (occurrence).



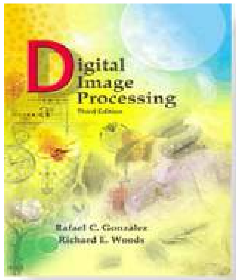
Examples of variable length encoding

r_k	$p_r(r_k)$	Code 1	$l_1(r_k)$	Code 2	$l_2(r_k)$
$r_{87} = 87$	0.25	01010111	8	01	2
$r_{128} = 128$	0.47	10000000	8	1	1
$r_{186} = 186$	0.25	11000100	8	000	3
$r_{255} = 255$	0.03	11111111	8	001	3
r_k for $k \neq 87, 128, 186, 255$	0	—	8	—	0



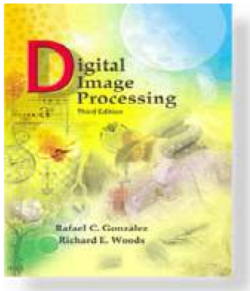
Spatial/Temopral Redundancy

- Internal Correlation between the pixel result from
  - Respective Autocorrelation
  - Structural Relationship
  - Geometric Relation ship
- The value of a pixel can be reasonably predicted from the values of its neighbors.
- To reduce the inter-pixel redundancies in an image the 2D array is transformed (*mapped*) into more efficient format (Frequency Domain etc.)



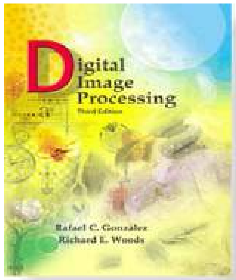
Irrelevant information and Psycho-Visual Redundancy

- The brightness of a region depend on other factors that the light reflection
- The perceived intensity of the eye is limited an non linear
- Certain information has less relative importance that other information in normal visual processing
- In general, observer searches for distinguishing features such as edges and textural regions.



Measuring Information

- A random even E that occurs with probability P(E) is said to contain I(E) information where I(E) is defined as
  $I(E) = \log(1/P(E)) = -\log(P(E))$
- $P(E) = 1$ contain no information
- $P(E) = ½$ requires one bit of information.



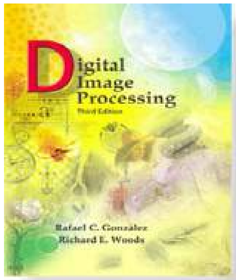
Measuring Information

- For a source of events a_0, a_1,a_2, .., a_k with associated probability $P(a_0)$, $P(a_1)$, $P(a_2)$, .., $P(a_k)$.
- The average information per source (entropy) is

$$H = -\sum_{j=0}^{k} P(a_j)\log(P(a_j))$$

For image, we use the normalized histogram to generate the source probability, which leas to the entropy

$$\tilde{H} = -\sum_{i=0}^{L-1} p_r(r_i)\log(p_r(r_i))$$



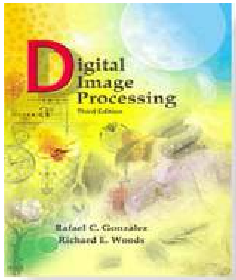
Fidelity Criteria

- Objective Fidelity Criteria
  - The information loss can be expressed as a function of the encoded and decoded images.
  - For image I (x,y) and its decoded approximation I'(x,y)
  - For any value of x and y, the error e(x,y) could be defined as

$$e(x,y) = I'(x,y) - I(x,y)$$

  - For the entire Image

$$\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} I'(x,y) - I(x,y)$$



Fidelity Criteria

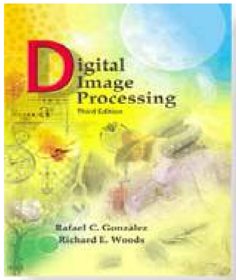
- The mean-square-error, e_{rms} is

$$e_{rms} = \sqrt{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[I'(x,y) - I(x,y)\right]^2}$$

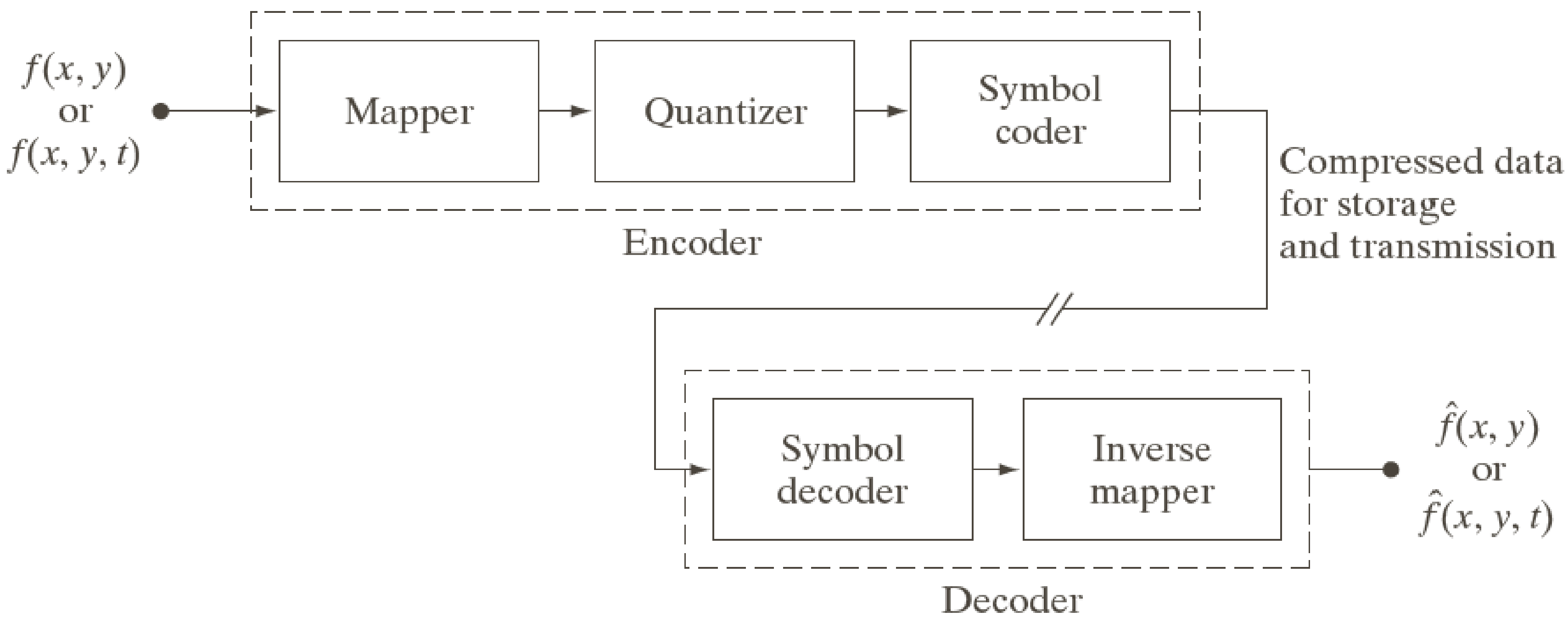
The mean-square-error signal-to-noise ratio SNR_{ms} is

$$SNR_{ms} = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} I'(x,y)^2}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[I'(x,y) - I(x,y)\right]^2}$$

Value	Rating	Description
1	Excellent	An image of extremely high quality, as good as you could desire.
2	Fine	An image of high quality, providing enjoyable viewing. Interference is not objectionable.
3	Passable	An image of acceptable quality. Interference is not objectionable.
4	Marginal	An image of poor quality; you wish you could improve it. Interference is somewhat objectionable.
5	Inferior	A very poor image, but you could watch it. Objectionable interference is definitely present.
6	Unusable	An image so bad that you could not watch it.



Three approximations of the same image


$f(x, y)$
or
$f(x, y, t)$
Mapper
Quantizer
Symbol coder
Encoder
Compressed data for storage and transmission
Symbol decoder
Inverse mapper
Decoder
$\hat{f}(x, y)$
or
$\hat{f}(x, y, t)$

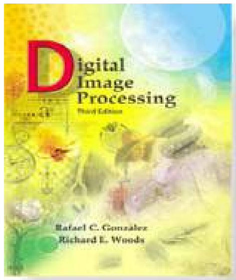


Image Compression
Standards, Formats, and Containers

Still Image

Binary

CCITT Group 3
CCITT Group 4
JBIG (or JBIG1)
JBIG2

TIFF

Continuous Tone

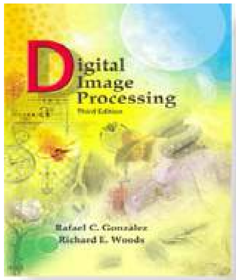
JPEG
JPEG-LS
JPEG-2000

BMP
GIF
PDF
PNG
TIFF

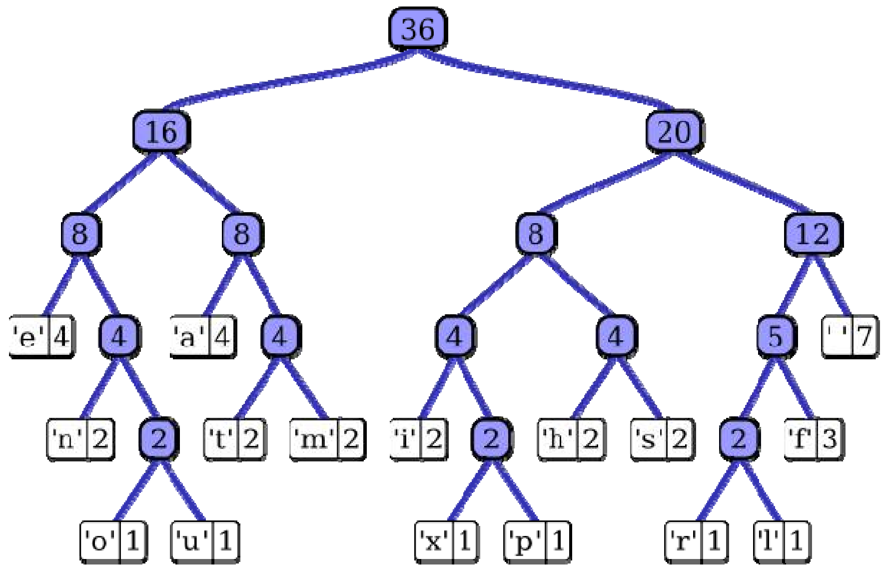
Video

DV
H.261
H.262
H.263
H.264
MPEG-1
MPEG-2
MPEG-4
MPEG-4 AVC

AVS
HDV
M-JPEG
QuickTime
VC-1 (or WMV9)



Huffman coding is an entropy encoding algorithm used for lossless data compression. The term refers to the use of a variable-length code table for encoding a source symbol (such as a character in a file) where the variable-length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol.

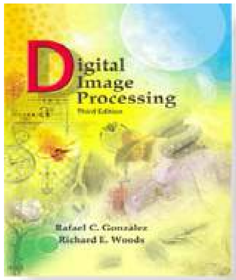


Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	0.4
a_1	0.1	0.1	0.2	0.3	
a_4	0.1	0.1	0.1		
a_3	0.06	0.1			
a_5	0.04				

Huffman coding

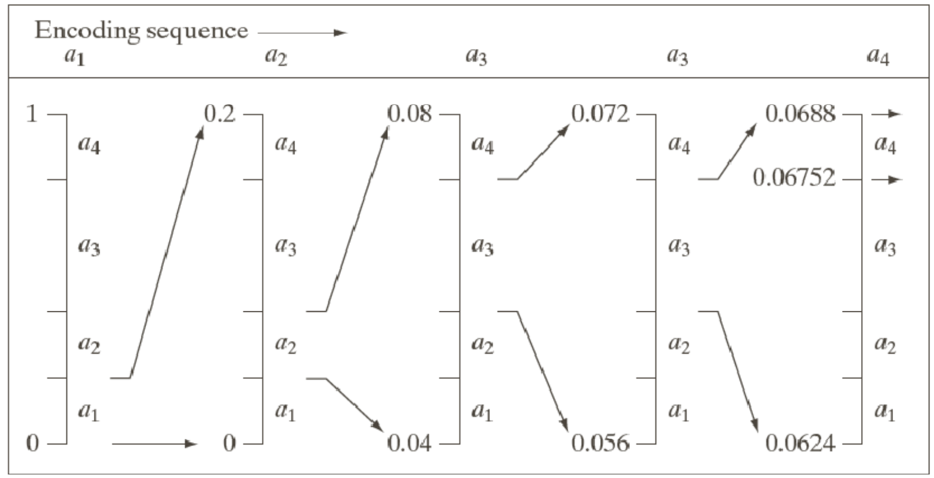
Assignment procedure

Original source			Source reduction			
Symbol	Probability	Code	1	2	3	4
a_2	0.4	1	0.4 1	0.4 1	0.4 1	0.6 0
a_6	0.3	00	0.3 00	0.3 00	0.3 00	0.4 1
a_1	0.1	011	0.1 011	0.2 010	0.3 01	
a_4	0.1	0100	0.1 0100	0.1 011		
a_3	0.06	01010	0.1 0101			
a_5	0.04	01011				

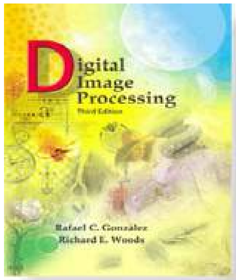


Arithmetic coding is a form of variable-length entropy encoding. A string is converted to arithmetic encoding, usually characters are stored with fewer bits

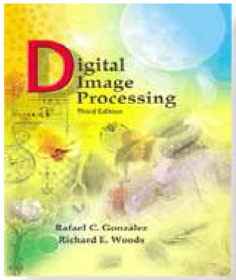
Arithmetic coding encodes the entire message into a single number, a fraction n where $(0.0 \leq n < 1.0)$.



Source Symbol	Probability	Initial Subinterval
a_1	0.2	[0.0, 0.2)
a_2	0.2	[0.2, 0.4)
a_3	0.4	[0.4, 0.8)
a_4	0.2	[0.8, 1.0)



Compression Algorithms



Symbol compression

This approaches determine a set of symbols that constitute the image, and take advantage of their multiple appearance. It convert each symbol into token, generate a token table and represent the compressed image as a list of tokens.

This approach is good for document images.

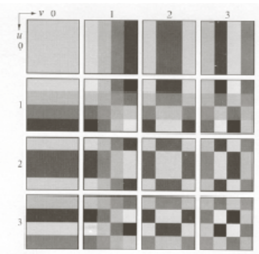


FIGURE 8.30 Discrete-cosine basis functions for $N = 4$. The origin of each block is at its top left.

where

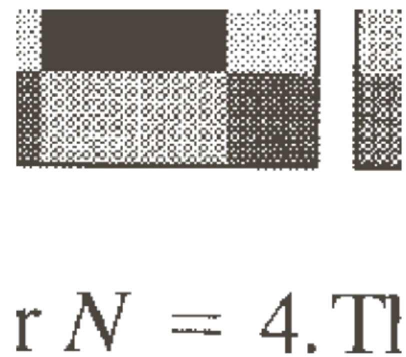
$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } u = 0 \\ \sqrt{\frac{2}{N}} & \text{for } u = 1, 2, \ldots, N-1 \end{cases} \quad (8.5\text{-}33)$$

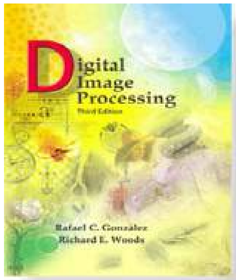
and similarly for $\alpha(v)$. Figure 8.30 shows $g(x, y, u, v)$ for the case $N = 4$. The computation follows the same format as explained for Fig. 8.29, with the difference that the values of g are not integers. In Fig. 8.30, the lighter gray levels correspond to larger values of g.

EXAMPLE 8.19: Transform coding with the DFT, WHT, and DCT.

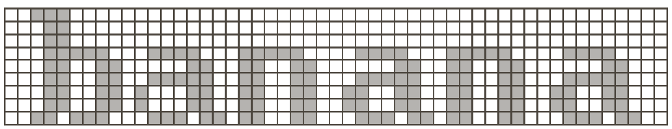
Figures 8.31(a), (c), and (e) show three approximations of the 512×512 monochrome image in Fig. 8.23. These pictures were obtained by dividing the original image into subimages of size 8×8, representing each subimage using one of the transforms just described (i.e., the DFT, WHT, or DCT transform), truncating 50% of the resulting coefficients, and taking the inverse transform of the truncated coefficient arrays.

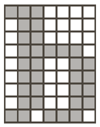
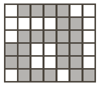
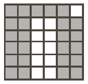
In each case, the 32 retained coefficients were selected on the basis of maximum magnitude. When we disregard any quantization or coding issues, this process amounts to compressing the original image by a factor of 2. Note that in all cases, the 32 discarded coefficients had little visual impact on reconstructed image quality. Their elimination, however, was accompanied by some mean-square error, which can be seen in the scaled error images of Figs. 8.31(b), (d), and (f). The actual rms errors were 1.28, 0.86, and 0.68 gray levels, respectively.





Chapter 8
Image Compression

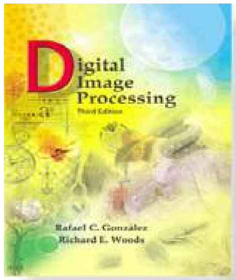


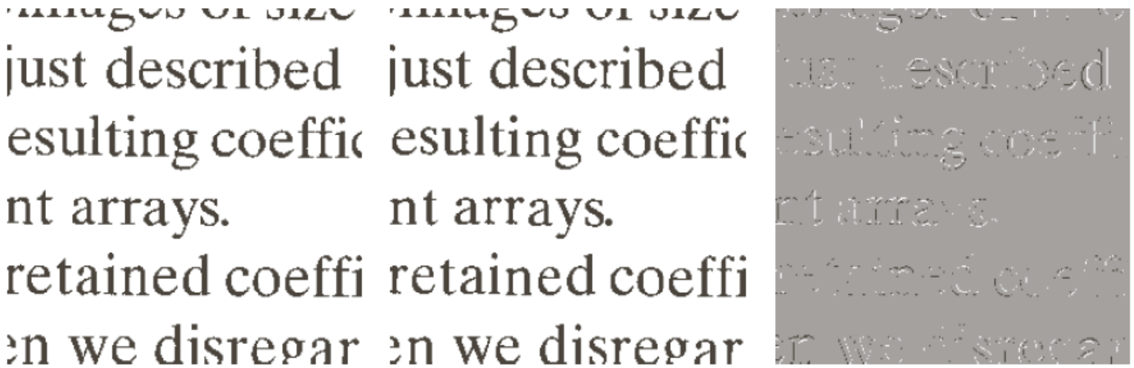
Token	Symbol
0	
1	
2	

Triplet
(0, 2, 0)
(3, 10, 1)
(3, 18, 2)
(3, 26, 1)
(3, 34, 2)
(3, 42, 1)

a b c

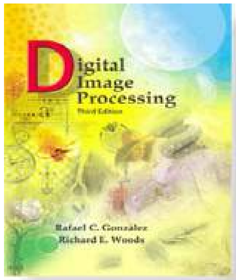
FIGURE 8.17 (a) A bi-level document, (b) symbol dictionary, and (c) the triplets used to locate the symbols in the document.





a b c

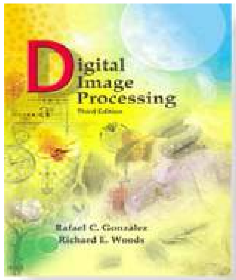
FIGURE 8.18 JBIG2 compression comparison: (a) lossless compression and reconstruction; (b) perceptually lossless; and (c) the scaled difference between the two.

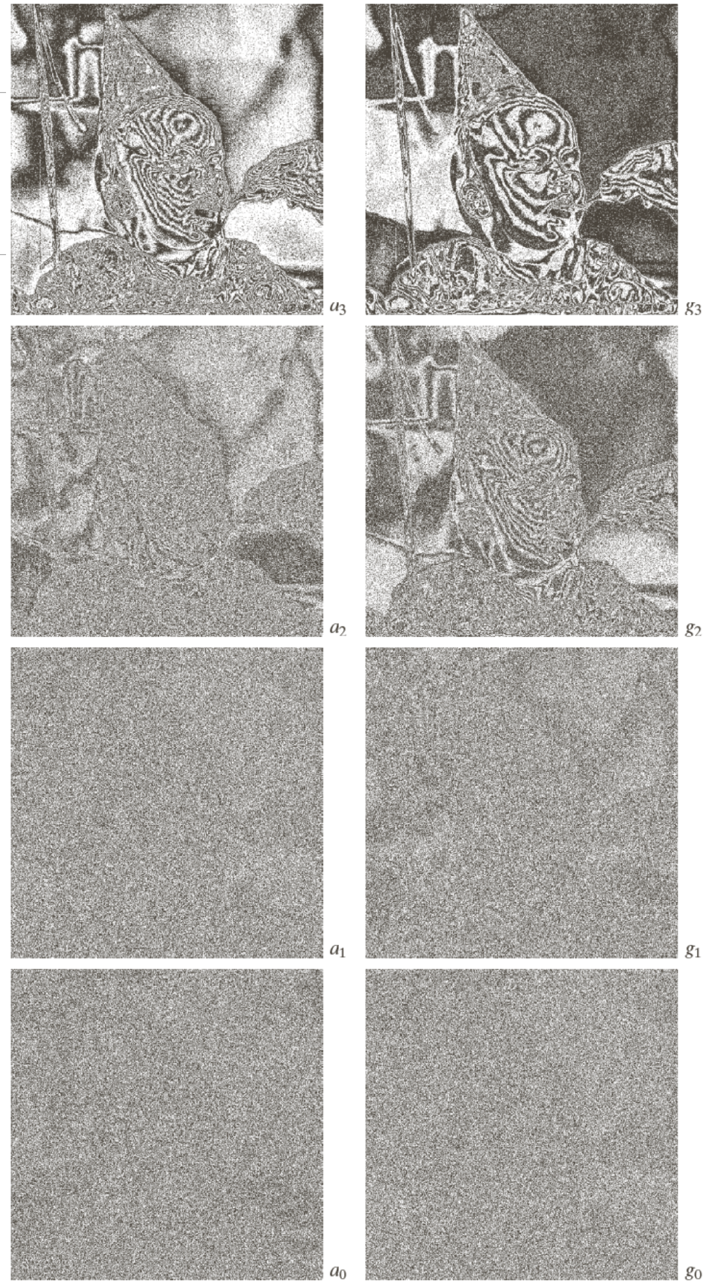


Digital Image Processing, 3rd ed.

a b
c d
e f
g h

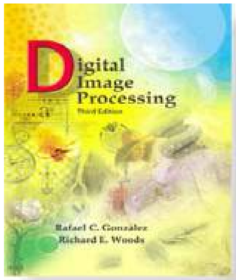
FIGURE 8.19
(a) A 256-bit monochrome image. (b)–(h) The four most significant binary and Gray-coded bit planes of the image in (a).

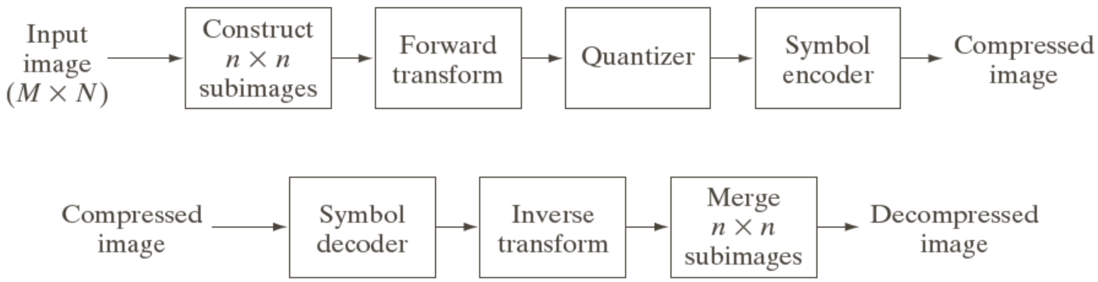




a	b
c	d
e	f
g	h

FIGURE 8.20 (a)–(h) The four least significant binary (left column) and Gray-coded (right column) bit planes of the image in Fig. 8.19(a).





a
b

FIGURE 8.21 A block transform coding system: (a) encoder; (b) decoder.

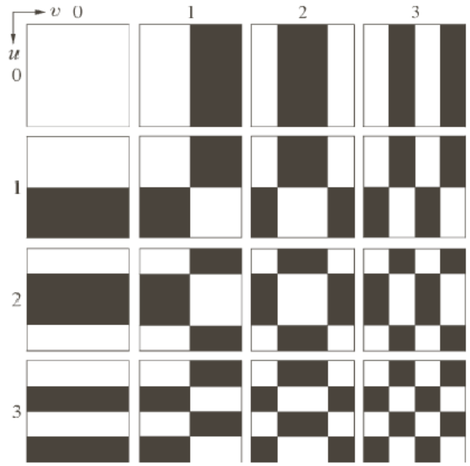
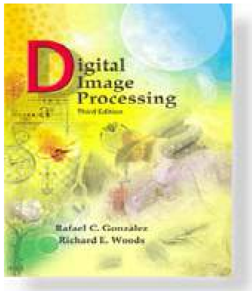


FIGURE 8.22 Walsh-Hadamard basis functions for $n = 4$. The origin of each block is at its top left.



Chapter 8
Image Compression

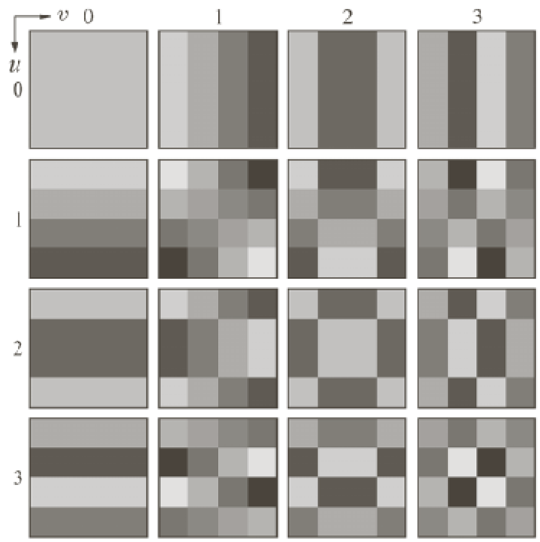
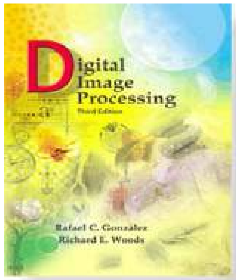


FIGURE 8.23 Discrete-cosine basis functions for $n = 4$. The origin of each block is at its top left.

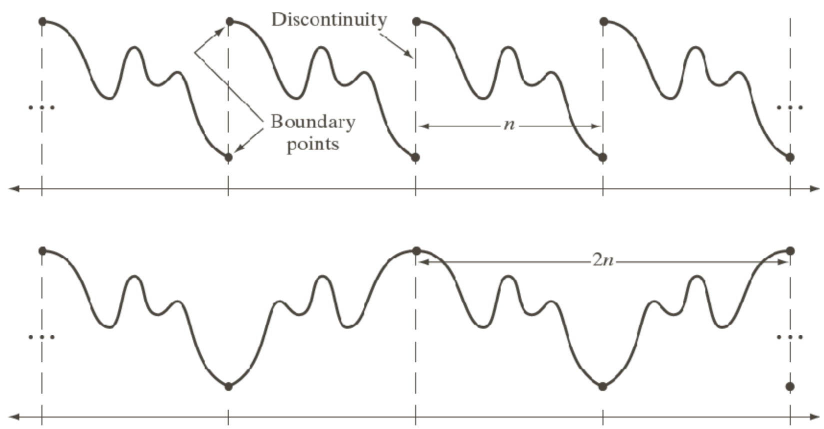
a b c
d e f

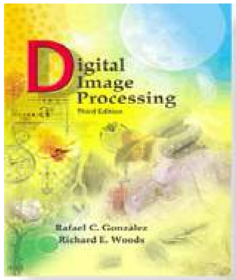
FIGURE 8.24 Approximations of Fig. 8.9(a) using the (a) Fourier, (b) Walsh-Hadamard, and (c) cosine transforms, together with the corresponding scaled error images in (d)–(f).



DFT and DCT

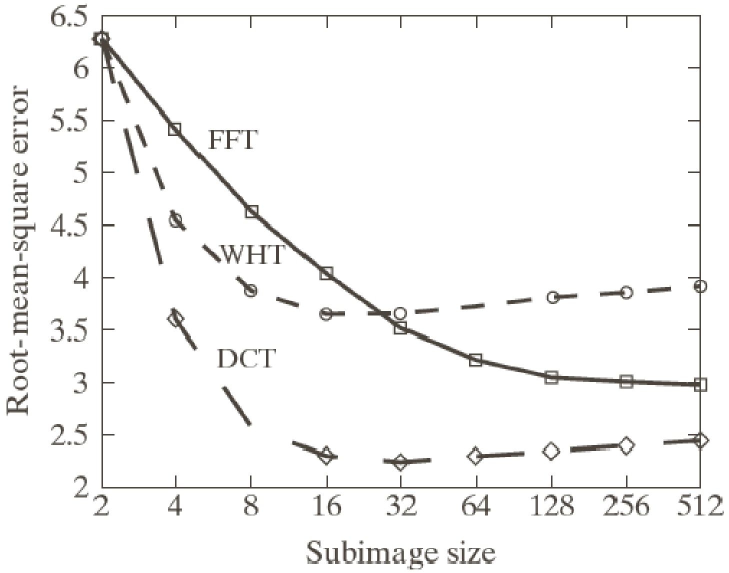
The periodicity implicit in the 1-D DFT and DCT. The DCT provide better continuity that the general DFT.

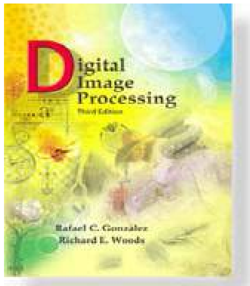




Bock Size vs. Reconstruction Error

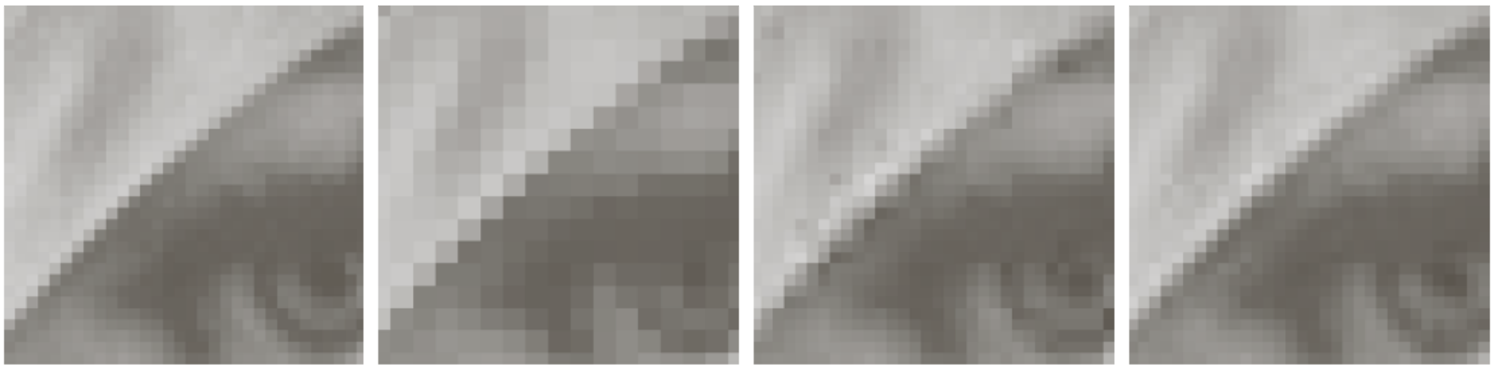
The DCT provide the least error at almost any sub-image size.
The error takes its minimum at sub-images of sizes between 16 and 32.





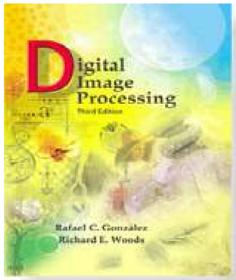
Chapter 8

Image Compression



a b c d

FIGURE 8.27 Approximations of Fig. 8.27(a) using 25% of the DCT coefficients and (b) 2 × 2 subimages, (c) 4 × 4 subimages, and (d) 8 × 8 subimages. The original image in (a) is a zoomed section of Fig. 8.9(a).

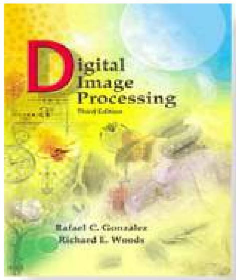


Chapter 8

Image Compression

a b
c d

FIGURE 8.28 Approximations of Fig. 8.9(a) using 12.5% of the 8×8 DCT coefficients: (a)–(b) threshold coding results; (c)–(d) zonal coding results. The difference images are scaled by 4.



1	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0
1	1	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

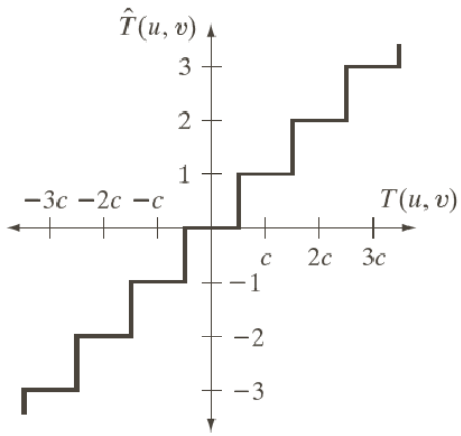
8	7	6	4	3	2	1	0
7	6	5	4	3	2	1	0
6	5	4	3	3	1	1	0
4	4	3	3	2	1	0	0
3	3	3	2	1	1	0	0
2	2	1	1	1	0	0	0
1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0

1	1	0	1	1	0	0	0
1	1	1	1	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	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

a b
c d

FIGURE 8.29 A typical (a) zonal mask, (b) zonal bit allocation, (c) threshold mask, and (d) thresholded coefficient ordering sequence. Shading highlights the coefficients that are retained.



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

a b

FIGURE 8.30 (a) A threshold coding quantization curve [see Eq. (8.2-29)]. (b) A typical normalization matrix.

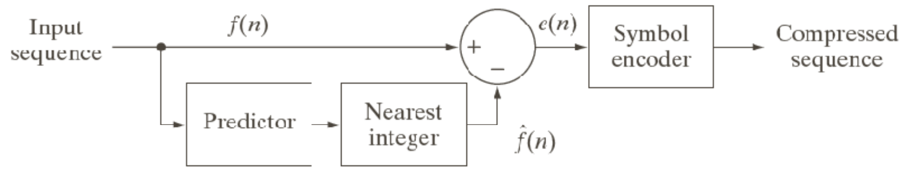
Chapter 8
Image Compression

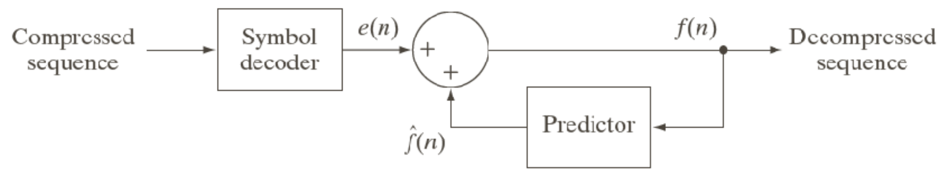
FIGURE 8.31 Approximations of Fig. 8.9(a) using the DCT and normalization array of Fig. 8.30(b): (a) **Z**, (b) 2**Z**, (c) 4**Z**, (d) 8**Z**, (e) 16**Z**, and (f) 32**Z**.

a b c
d e f

FIGURE 8.32 Two JPEG approximations of Fig. 8.9(a). Each row contains a result after compression and reconstruction, the scaled difference between the result and the original image, and a zoomed portion of the reconstructed image.

Lossless Predictive coding





The encoder expects a discrete samples of a signal f(n).

1. A predictor is applied and its output is rounded to the nearest integer. $\hat{f}(n)$
2. The error is estimated as $e(n) = f(n) - \hat{f}(n)$
3. The compressed stream consist of first sample and the errors, encoded using variable length coding

The decoder uses the predictor and the error stream to reconstructs the original signal f(n).

1. The predictor is initialized using the first sample.
2. The received error is added to predictor result.

$$f(n) = \hat{f}(n) + e(n)$$

Lossless Predictive coding

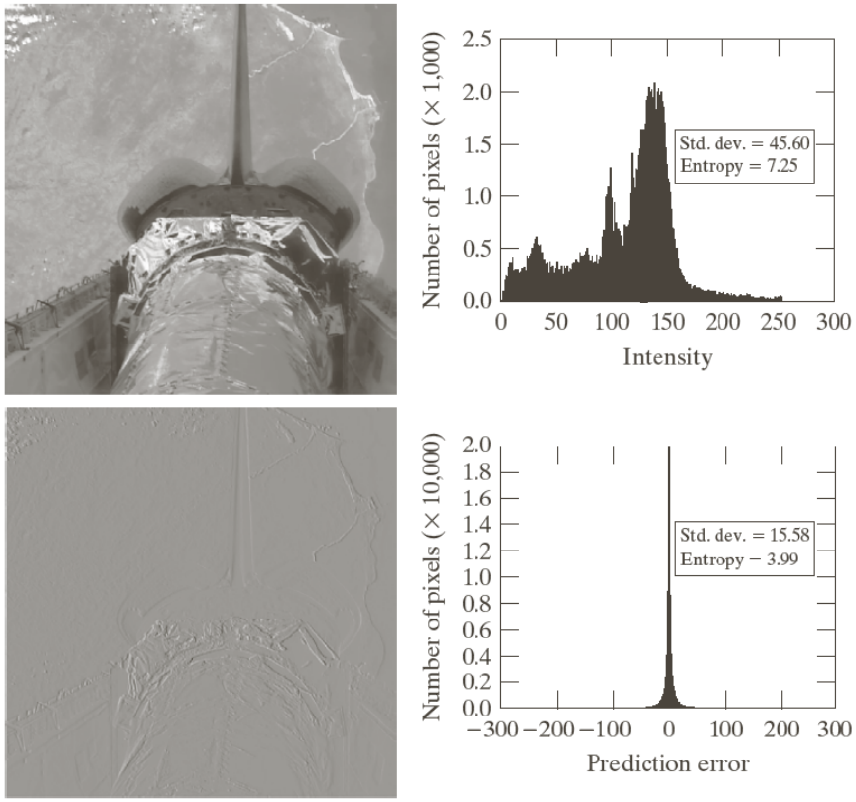
Linear predictors usually have the form:

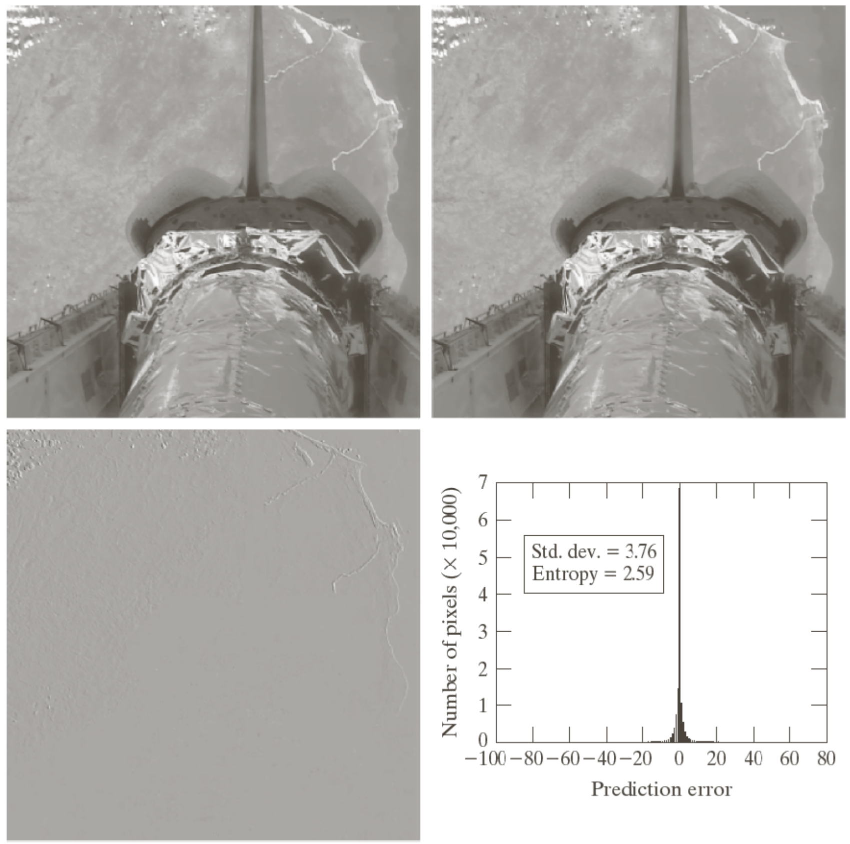
$$\hat{f}(n) = round\left[\sum_{i=0}^{m} a_i f(n-i)\right]$$

Original Image (view of the earth).

The prediction error and its histogram.

1. The error is small in uniform regions
2. Large close to edges and sharp changes in pixel intensities



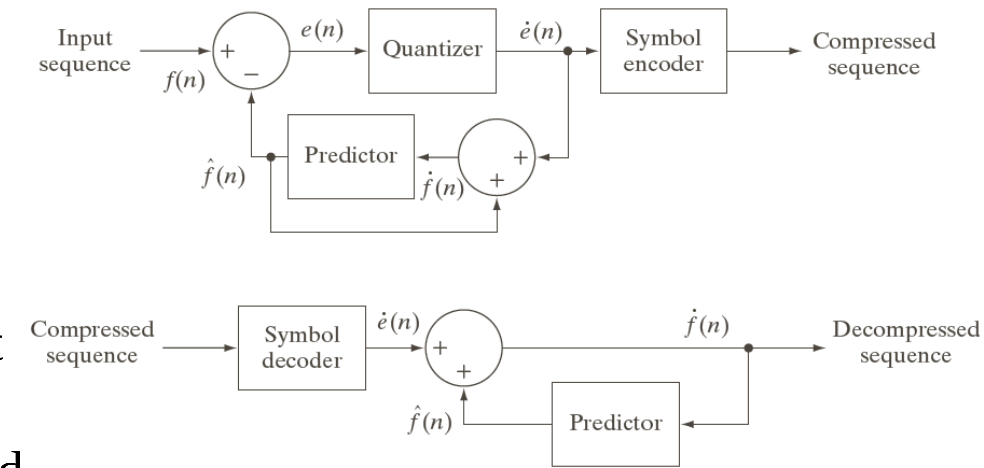


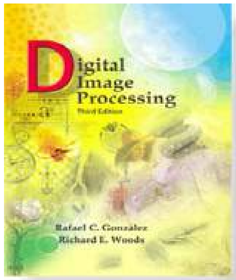
Number of pixels (× 10,000)
7
6
5
4
3
2
1
0
Std. dev. = 3.76
Entropy = 2.59
−100 −80 −60 −40 −20 0 20 40 60 80
Prediction error

Lossy Predictive coding

The encoder expects a discrete samples of a signal f(n).

1. A predictor is applied and its output is rounded to the nearest integer, $\hat{f}(n)$
2. The error is mapped into limited rage of values (quantized) $\dot{e}(n)$
3. The compressed stream consist of first sample and the mapped errors, encoded using variable length coding





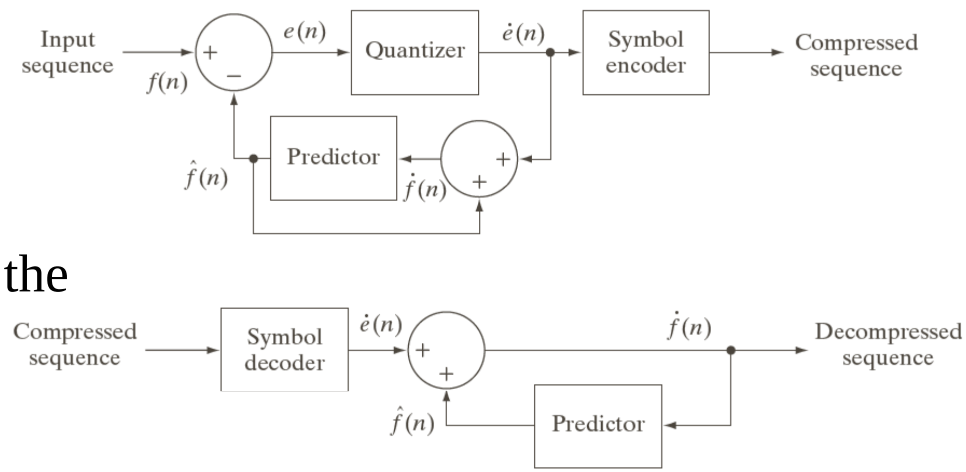
Lossy Predictive coding

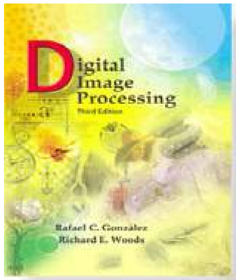
The decoder uses error stream to reconstructs an approximation of the original signal, $\dot{f}(n)$

1. The predictor is initialized using the first sample.
2. The received error is added to predictor result.

$$\dot{f}(n) = \dot{e}(n) + \hat{f}(n)$$

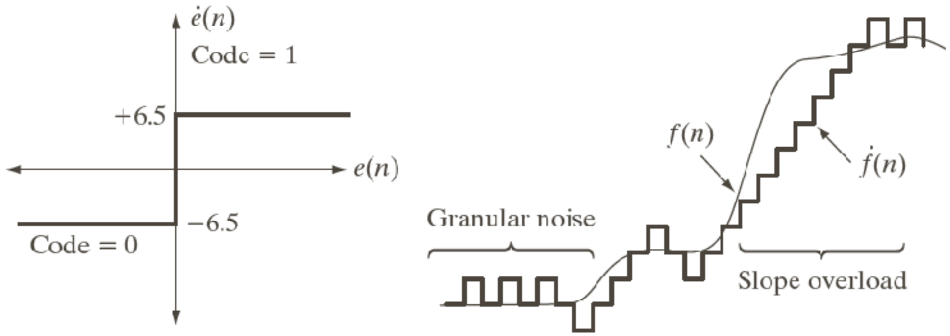
$$\dot{e}(n) = \begin{cases} +\xi & e(n) > 0 \\ -\xi & otherwise \end{cases}$$





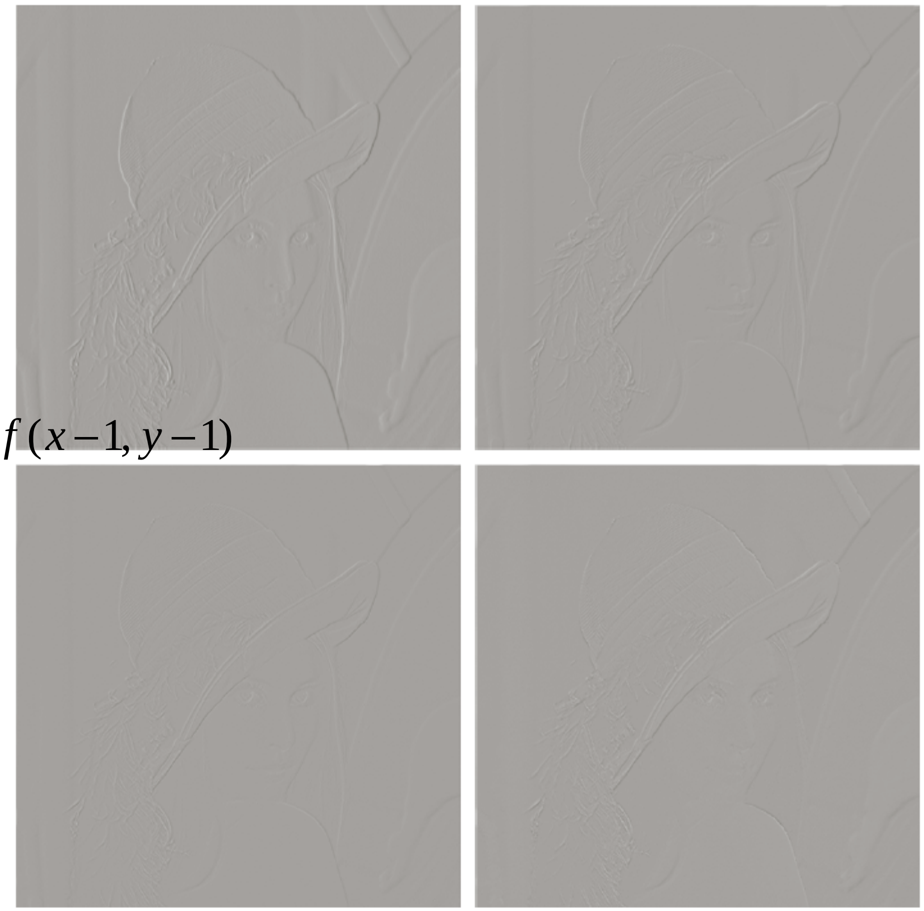
Chapter 8

Image Compression



Input		Encoder				Decoder		Error
n	$f(n)$	$\hat{f}(n)$	$e(n)$	$\dot{e}(n)$	$\dot{f}(n)$	$\hat{f}(n)$	$\dot{f}(n)$	$f(n) - \dot{f}(n)$
0	14	—	—	—	14.0	—	14.0	0.0
1	15	14.0	1.0	6.5	20.5	14.0	20.5	−5.5
2	14	20.5	−6.5	−6.5	14.0	20.5	14.0	0.0
3	15	14.0	1.0	6.5	20.5	14.0	20.5	−5.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
14	29	20.5	8.5	6.5	27.0	20.5	27.0	2.0
15	37	27.0	10.0	6.5	33.5	27.0	33.5	3.5
16	47	33.5	13.5	6.5	40.0	33.5	40.0	7.0
17	62	40.0	22.0	6.5	46.5	40.0	46.5	15.5
18	75	46.5	28.5	6.5	53.0	46.5	53.0	22.0
19	77	53.0	24.0	6.5	59.6	53.0	59.6	17.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

Prediction Error



The following images show the prediction error of the predictor

$$\hat{f}(x,y) = 0.97\, f(x, y-1)$$

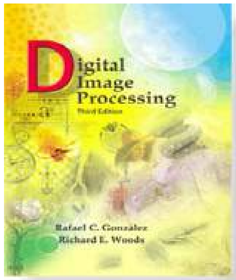
$$\hat{f}(x,y) = 0.5\, f(x, y-1) + 0.5\, f(x-1, y)$$

$$\hat{f}(x,y) = 0.75\, f(x, y-1) + 0.75\, f(x-1, y) - 0.5\, f(x-1, y-1)$$

$$\hat{f}(x,y) = \begin{cases} 0.97\, f(x, y-1) & \Delta h \le \Delta v \\ 0.97\, f(x-1, y) & otherwise \end{cases}$$

$$\Delta h = | f(x-1, y) - f(x-1, y-1)|$$

$$\Delta v = | f(x, y-1) - f(x-1, y-1)|$$



Optimal Predictors

What are the parameters of a linear predictor that minimize error

$$E\{e^2(n)\} = E\left\{\left[f(n) - \hat{f}(n)\right]^2\right\}$$

While taking into account

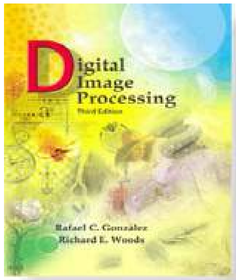
$$\dot{f}(n) = \dot{e}(n) + \hat{f}(n) \cong e(n) + \hat{f}(n) = f(n)$$

Using the definition of linear predictor

$$E\{e^2(n)\} = E\left\{\left[f(n) - \sum_{i=1}^{m} \alpha_i f(n-1)\right]^2\right\}$$

We assume that f(n) has a mean zero and variance σ^2

$$\alpha = R^{-1} r$$

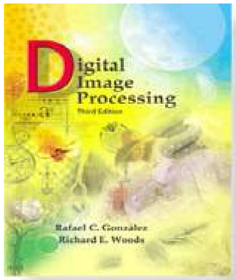


And R^{-1} is the mxn autocorrelation matrix

$$R = \begin{bmatrix} E\{f(n-1)f(n-1)\} & E\{f(n-1)f(n-2)\} & \dots & E\{f(n-1)f(n-m)\} \\ E\{f(n-2)f(n-1)\} & E\{f(n-2)f(n-2)\} & \dots & E\{f(n-2)f(n-m)\} \\ \vdots & \vdots & \vdots & \vdots \\ E\{f(n-m)f(n-1)\} & E\{f(n-m)f(n-1)\} & \dots & E\{f(n-m)f(n-1)\} \end{bmatrix}$$

$$r = \begin{bmatrix} E\{f(n)f(n-1)\} \\ \vdots \\ E\{f(n-1)f(n-m)\} \end{bmatrix}$$

$$a = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix}$$



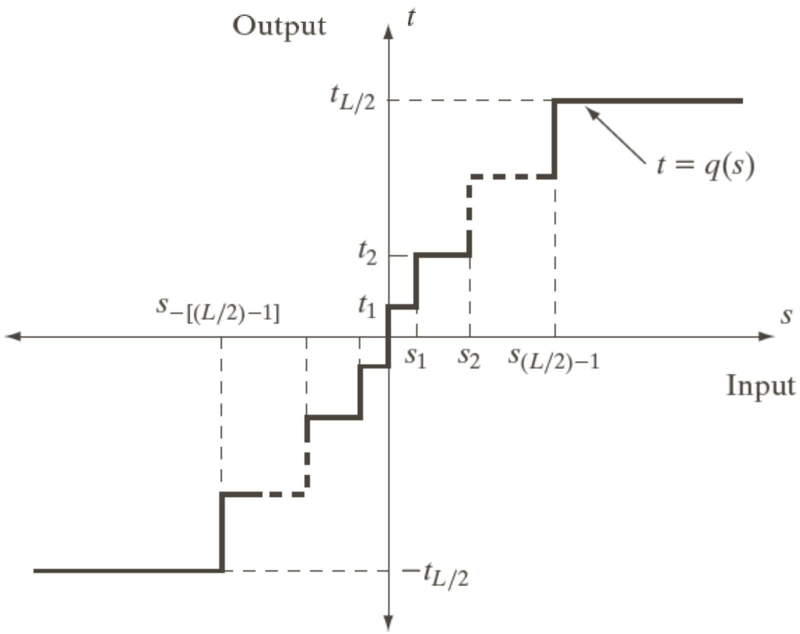
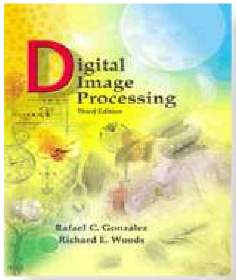
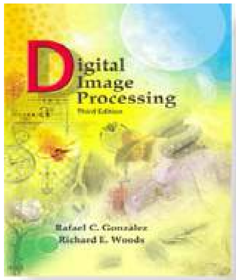


FIGURE 8.44 A typical quantization function.

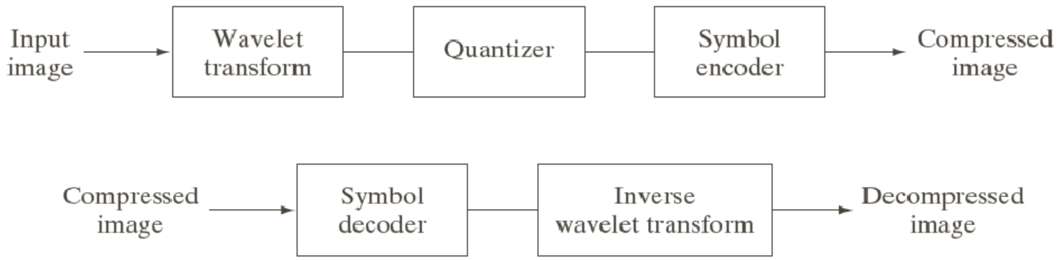


Levels	**2**		**4**		**8**	
i	s_i	t_i	s_i	t_i	s_i	t_i
1	∞	0.707	1.102	0.395	0.504	0.222
2			∞	1.810	1.181	0.785
3					2.285	1.576
4					∞	2.994
θ	1.414		1.087		0.731	

TABLE 8.12
Lloyd-Max quantizers for a Laplacian probability density function of unit variance.

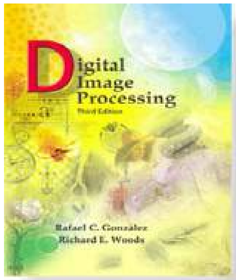


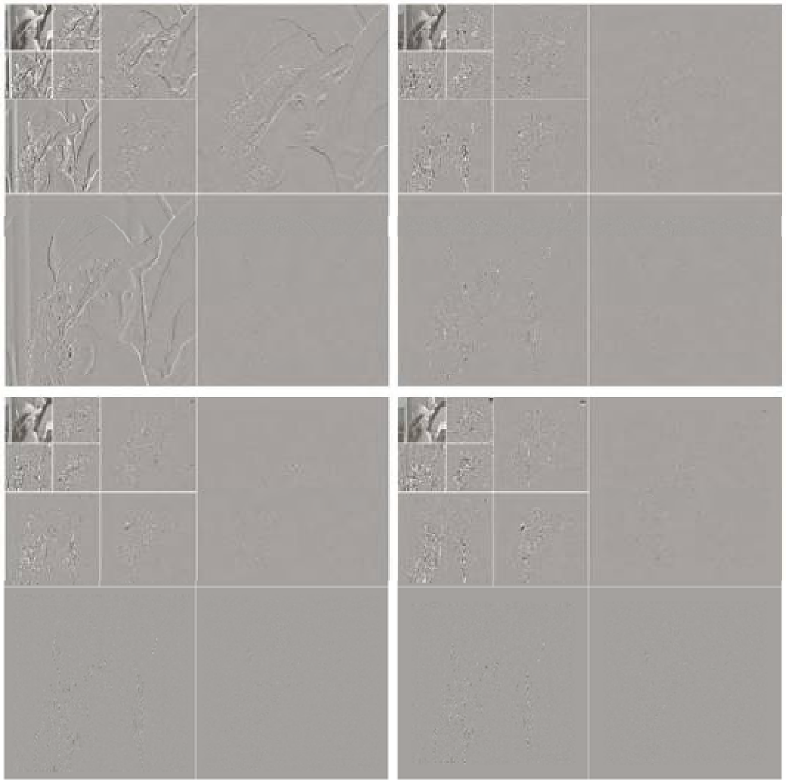
Chapter 8
Image Compression



a
b

FIGURE 8.45 A wavelet coding system: (a) encoder; (b) decoder.



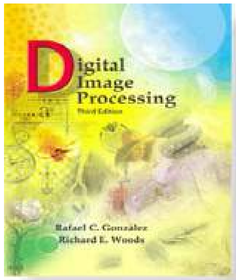


a b
c d

FIGURE 8.46 Three-scale wavelet transforms of Fig. 8.9(a) with respect to (a) Haar wavelets, (b) Daubechies wavelets, (c) symlets, and (d) Cohen-Daubechies Feauveau biorthogonal wavelets.

Wavelet	Filter Taps (Scaling + Wavelet)	Zeroed Coefficients
Haar (see Ex. 7.10)	2 + 2	33.8%
Daubechies (see Fig. 7.8)	8 + 8	40.9%
Symlet (see Fig. 7.26)	8 + 8	41.2%
Biorthogonal (see Fig. 7.39)	17 + 11	42.1%

TABLE 8.13 Wavelet transform filter taps and zeroed coefficients when truncating the transforms in Fig. 8.46 below 1.5.



Decomposition Level (Scales or Filter Bank Iterations)	Approximation Coefficient Image	Truncated Coefficients (%)	Reconstruction Error (rms)
1	256 × 256	74.7%	3.27
2	128 × 128	91.7%	4.23
3	64 × 64	95.1%	4.54
4	32 × 32	95.6%	4.61
5	16 × 16	95.5%	4.63

TABLE 8.14 Decomposition level impact on wavelet coding the 512 × 512 image of Fig. 8.9(a).

Chapter 8
Image Compression

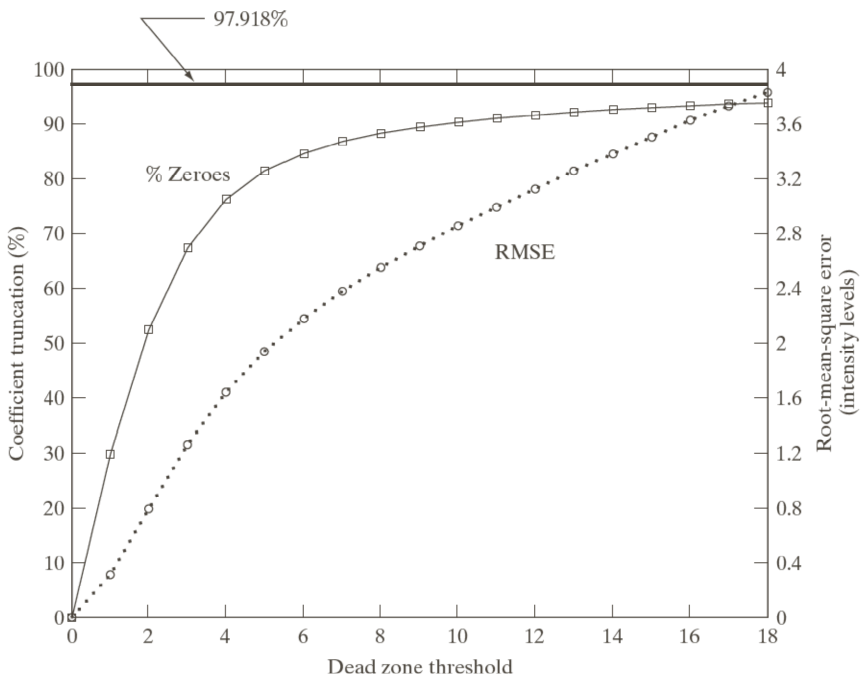
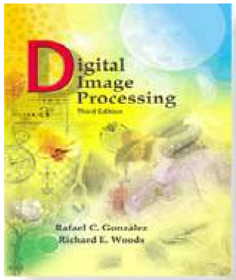


FIGURE 8.47 The impact of dead zone interval selection on wavelet coding.



Filter Tap	Highpass Wavelet Coefficient	Lowpass Scaling Coefficient
0	−1.115087052456994	0.6029490182363579
±1	0.5912717631142470	0.2668641184428723
±2	0.05754352622849957	−0.07822326652898785
±3	−0.09127176311424948	−0.01686411844287495
±4	0	0.02674875741080976

TABLE 8.15 Impulse responses of the low- and highpass analysis filters for an irreversible 9-7 wavelet transform.

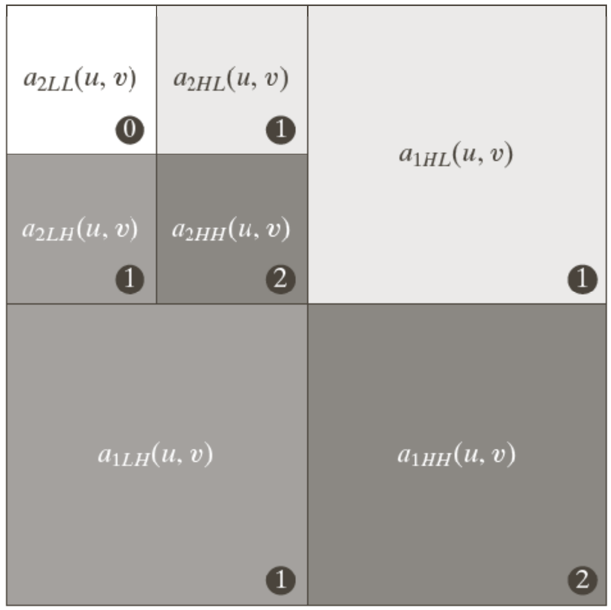


FIGURE 8.48 JPEG 2000 two-scale wavelet transform tile-component coefficient notation and analysis gain.

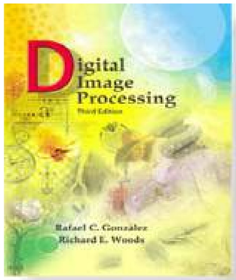


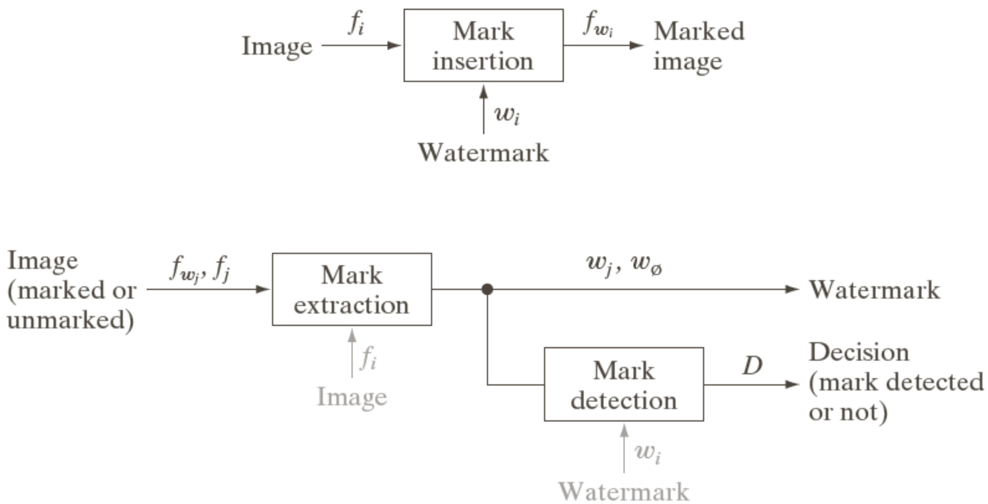
FIGURE 8.49 Four JPEG-2000 approximations of Fig. 8.9(a). Each row contains a result after compression and reconstruction, the scaled difference between the result and the original image, and a zoomed portion of the reconstructed image. (Compare the results in rows 1 and 2 with the JPEG results in Fig. 8.32.)

a
b c

FIGURE 8.50 A simple visible watermark: (a) watermark; (b) the watermarked image; and (c) the difference between the watermarked image and the original (non-watermarked) image.

a b
c d

FIGURE 8.51 A simple invisible watermark: (a) watermarked image; (b) the extracted watermark; (c) the watermarked image after high quality JPEG compression and decompression; and (d) the extracted watermark from (c).



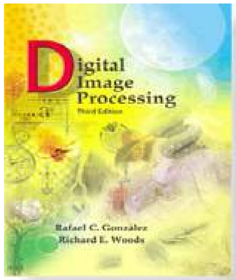
a
b

FIGURE 8.52 A typical image watermarking system: (a) encoder; (b) decoder.

Chapter 8
Image Compression

a b
c d

FIGURE 8.53 (a) and (c) Two watermarked versions of Fig. 8.9(a); (b) and (d) the differences (scaled in intensity) between the watermarked versions and the unmarked image. These two images show the intensity contribution (although scaled dramatically) of the pseudo-random watermarks on the original image.



$\gamma = 0.9945$ $\gamma = 0.7395$ $\gamma = 0.8390$

$\gamma = 0.8230$ $\gamma = 0.5210$ $\gamma = 0.3113$

a b c
d e f

FIGURE 8.54 Attacks on the watermarked image in Fig. 8.53(a): (a) lossy JPEG compression and decompression with an rms error of 7 intensity levels; (b) lossy JPEG compression and decompression with an rms error of 10 intensity levels (note the blocking artifact); (c) smoothing by spatial filtering; (d) the addition of Gaussian noise; (e) histogram equalization; and (f) rotation. Each image is a modified version of the watermarked image in Fig. 8.53(a). After modification, they retain their watermarks to varying degrees, as indicated by the correlation coefficients below each image.