The size of typical still image (1200x1600)

\[ 1200 \times 1600 \times 3\text{byte} = 5760000\text{byte} \]

\[ = 5,760\text{Kbyte} = 5.76\text{Mbyte} \]

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

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

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

\[ = 224\text{GByte}. \]
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
  \[ R = 1 - \frac{1}{C} \]
  where $C = \frac{n1}{n2}$
• 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.
Chapter 8
Image Compression

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

<table>
<thead>
<tr>
<th>$r_k$</th>
<th>$p_r(r_k)$</th>
<th>Code 1</th>
<th>$l_1(r_k)$</th>
<th>Code 2</th>
<th>$l_2(r_k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{87} = 87$</td>
<td>0.25</td>
<td>01010111</td>
<td>8</td>
<td>01</td>
<td>2</td>
</tr>
<tr>
<td>$r_{128} = 128$</td>
<td>0.47</td>
<td>10000000</td>
<td>8</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$r_{186} = 186$</td>
<td>0.25</td>
<td>11000100</td>
<td>8</td>
<td>000</td>
<td>3</td>
</tr>
<tr>
<td>$r_{255} = 255$</td>
<td>0.03</td>
<td>11111111</td>
<td>8</td>
<td>001</td>
<td>3</td>
</tr>
<tr>
<td>$r_k$ for $k \neq 87, 128, 186, 255$</td>
<td>0</td>
<td>—</td>
<td>8</td>
<td>—</td>
<td>0</td>
</tr>
</tbody>
</table>
Spatial/Temopral Redundancy

- Internal Correlation between the pixel result from
  - Respective Autocorrelation
  - Structural Relationship
  - Geometric Relationship

- 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 event 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) = \(\frac{1}{2}\) requires one bit of information.
Measuring Information

• For a source of events $a_0, a_1, a_2, \ldots, a_k$ with associated probability $P(a_0), P(a_1), P(a_2), \ldots, P(a_k)$.

• 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 leads 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

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

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

$$SNR_{ms} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I'(x, y)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [I'(x, y) - I(x, y)]^2}$$
# Chapter 8: Image Compression

<table>
<thead>
<tr>
<th>Value</th>
<th>Rating</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Excellent</td>
<td>An image of extremely high quality, as good as you could desire.</td>
</tr>
<tr>
<td>2</td>
<td>Fine</td>
<td>An image of high quality, providing enjoyable viewing.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interference is not objectionable.</td>
</tr>
<tr>
<td>3</td>
<td>Passable</td>
<td>An image of acceptable quality. Interference is not objectionable.</td>
</tr>
<tr>
<td>4</td>
<td>Marginal</td>
<td>An image of poor quality; you wish you could improve it.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interference is somewhat objectionable.</td>
</tr>
<tr>
<td>5</td>
<td>Inferior</td>
<td>A very poor image, but you could watch it. Objectionable interference is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>definitely present.</td>
</tr>
<tr>
<td>6</td>
<td>Unusable</td>
<td>An image so bad that you could not watch it.</td>
</tr>
</tbody>
</table>
Three approximations of the same image
Chapter 8
Image Compression
Chapter 8
Image Compression

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

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

BMP
GIF
PDF
PNG
TIFF

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.
### Huffman coding

Assignment procedure

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Probability</th>
<th>Code</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_2$</td>
<td>0.4</td>
<td>1</td>
<td>0.4</td>
<td>1</td>
<td>0.4</td>
<td>1</td>
</tr>
<tr>
<td>$a_6$</td>
<td>0.3</td>
<td>00</td>
<td>0.3</td>
<td>00</td>
<td>0.3</td>
<td>00</td>
</tr>
<tr>
<td>$a_1$</td>
<td>0.1</td>
<td>011</td>
<td>0.1</td>
<td>011</td>
<td>0.2</td>
<td>010</td>
</tr>
<tr>
<td>$a_4$</td>
<td>0.1</td>
<td>0100</td>
<td>0.1</td>
<td>0100</td>
<td>0.1</td>
<td>011</td>
</tr>
<tr>
<td>$a_3$</td>
<td>0.06</td>
<td>01010</td>
<td>0.1</td>
<td>0101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$a_5$</td>
<td>0.04</td>
<td>01011</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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 \( 0.0 \leq n < 1.0 \).
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.

\[
N = 4
\]
Chapter 8
Image Compression

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

<table>
<thead>
<tr>
<th>Token</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><img src="image" alt="Symbol 0" /></td>
</tr>
<tr>
<td>1</td>
<td><img src="image" alt="Symbol 1" /></td>
</tr>
<tr>
<td>2</td>
<td><img src="image" alt="Symbol 2" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Triplet</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 2, 0)</td>
</tr>
<tr>
<td>(3, 10, 1)</td>
</tr>
<tr>
<td>(3, 18, 2)</td>
</tr>
<tr>
<td>(3, 26, 1)</td>
</tr>
<tr>
<td>(3, 34, 2)</td>
</tr>
<tr>
<td>(3, 42, 1)</td>
</tr>
</tbody>
</table>
Images of size $M \times N$ just described. The resulting coefficients are retained in arrays. We disregard coefficients that we disregard.

**FIGURE 8.18**

JBIG compression comparison:
(a) lossless compression and reconstruction;
(b) perceptually lossless; and
(c) the scaled difference between the two.
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).
**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).

Chapter 8

Image Compression

![Diagram of image compression process]

**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.
Chapter 8
Image Compression

**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

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

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.
### Chapter 8
Image Compression

#### 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.

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8</th>
<th>7</th>
<th>6</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

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

**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) 2Z, (c) 4Z, (d) 8Z, (e) 16Z, and (f) 32Z.
Chapter 8
Image Compression

**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 sample 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:

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

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
Chapter 8
Image Compression
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) $\hat{e}(n)$
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, \( \hat{f}(n) \)

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

\[
\hat{f}(n) = \hat{e}(n) + \hat{f}(n)
\]

\[
\hat{e}(n) = \begin{cases} 
+\xi & e(n) > 0 \\
-\xi & \text{otherwise}
\end{cases}
\]
# Chapter 8

## Image Compression

![Diagram showing image compression](image)

<table>
<thead>
<tr>
<th>Input</th>
<th>Encoder</th>
<th>Decoder</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>$f(n)$</td>
<td>$\hat{f}(n)$</td>
<td>$\hat{c}(n)$</td>
</tr>
<tr>
<td>0</td>
<td>14</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>14.0</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>20.5</td>
<td>-6.5</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>14.0</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>29</td>
<td>20.5</td>
<td>8.5</td>
</tr>
<tr>
<td>15</td>
<td>37</td>
<td>27.0</td>
<td>10.0</td>
</tr>
<tr>
<td>16</td>
<td>47</td>
<td>33.5</td>
<td>13.5</td>
</tr>
<tr>
<td>17</td>
<td>62</td>
<td>40.0</td>
<td>22.0</td>
</tr>
<tr>
<td>18</td>
<td>75</td>
<td>46.5</td>
<td>28.5</td>
</tr>
<tr>
<td>19</td>
<td>77</td>
<td>53.0</td>
<td>24.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prediction Error

The following images show the prediction error of the predictor

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

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

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

\[
\hat{f}(x, y) = \begin{cases} 
0.97 f(x, y - 1) & \Delta h \leq \Delta v \\
0.97 f(x - 1, y) & \text{otherwise}
\end{cases}
\]

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

\[
\Delta v = | f(x, y - 1) - f(x - 1, y - 1) |
\]
Optimal Predictors

What are the parameters of a linear predictor that minimize error 

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

While taking into account 

\[ \hat{f}(n) = \hat{e}(n) + \hat{f}(n) \approx e(n) + \hat{f}(n) = f(n) \]

Using the definition of linear predictor 

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

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

\[ \alpha = R^{-1}r \]
And \( R^{-1} \) is the mxn autocorrelation matrix

\[
R = \begin{bmatrix}
E\{f(n-1)f(n-1)\} & E\{f(n-1)f(n-2)\} & \ldots & E\{f(n-1)f(n-m)\} \\
E\{f(n-2)f(n-1)\} & E\{f(n-2)f(n-2)\} & \ldots & E\{f(n-2)f(n-m)\} \\
\vdots & \vdots & \ddots & \vdots \\
E\{f(n-m)f(n-1)\} & E\{f(n-m)f(n-1)\} & \ldots & E\{f(n-m)f(n-1)\}
\end{bmatrix}
\]

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

\[
a = \begin{bmatrix}
a_1 \\
\vdots \\
a_m
\end{bmatrix}
\]
FIGURE 8.44
A typical quantization function.
Chapter 8
Image Compression

<table>
<thead>
<tr>
<th>Levels</th>
<th>2</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i$</td>
<td>$s_i$</td>
<td>$t_i$</td>
<td>$s_i$</td>
</tr>
<tr>
<td>1</td>
<td>$\infty$</td>
<td>0.707</td>
<td>1.102</td>
</tr>
<tr>
<td>2</td>
<td>$\infty$</td>
<td>1.810</td>
<td>1.181</td>
</tr>
<tr>
<td>3</td>
<td>2.285</td>
<td>1.576</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$\infty$</td>
<td>2.994</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1.414</td>
<td>1.087</td>
<td>0.731</td>
</tr>
</tbody>
</table>

**TABLE 8.12**
Lloyd-Max quantizers for a Laplacian probability density function of unit variance.
Chapter 8
Image Compression

**FIGURE 8.45**
A wavelet coding system:
(a) encoder;
(b) decoder.
Chapter 8
Image Compression

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.
## Chapter 8

### Image Compression

<table>
<thead>
<tr>
<th>Wavelet</th>
<th>Filter Taps (Scaling + Wavelet)</th>
<th>Zeroed Coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>Haar (see Ex. 7.10)</td>
<td>2 + 2</td>
<td>33.8%</td>
</tr>
<tr>
<td>Daubechies (see Fig. 7.8)</td>
<td>8 + 8</td>
<td>40.9%</td>
</tr>
<tr>
<td>Symlet (see Fig. 7.26)</td>
<td>8 + 8</td>
<td>41.2%</td>
</tr>
<tr>
<td>Biorthogonal (see Fig. 7.39)</td>
<td>17 + 11</td>
<td>42.1%</td>
</tr>
</tbody>
</table>

**TABLE 8.13**

Wavelet transform filter taps and zeroed coefficients when truncating the transforms in Fig. 8.46 below 1.5.
## Image Compression

<table>
<thead>
<tr>
<th>Decomposition Level (Scales or Filter Bank Iterations)</th>
<th>Approximation Coefficient Image</th>
<th>Truncated Coefficients (%)</th>
<th>Reconstruction Error (rms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$256 \times 256$</td>
<td>74.7%</td>
<td>3.27</td>
</tr>
<tr>
<td>2</td>
<td>$128 \times 128$</td>
<td>91.7%</td>
<td>4.23</td>
</tr>
<tr>
<td>3</td>
<td>$64 \times 64$</td>
<td>95.1%</td>
<td>4.54</td>
</tr>
<tr>
<td>4</td>
<td>$32 \times 32$</td>
<td>95.6%</td>
<td>4.61</td>
</tr>
<tr>
<td>5</td>
<td>$16 \times 16$</td>
<td>95.5%</td>
<td>4.63</td>
</tr>
</tbody>
</table>

**TABLE 8.14**
Decomposition level impact on wavelet coding the $512 \times 512$ image of Fig. 8.9(a).
Chapter 8
Image Compression

**Figure 8.47** The impact of dead zone interval selection on wavelet coding.
### Chapter 8
Image Compression

<table>
<thead>
<tr>
<th>Filter Tap</th>
<th>Highpass Wavelet Coefficient</th>
<th>Lowpass Scaling Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-1.115087052456994</td>
<td>0.6029490182363579</td>
</tr>
<tr>
<td>±1</td>
<td>0.5912717631142470</td>
<td>0.2668641184428723</td>
</tr>
<tr>
<td>±2</td>
<td>0.05754352622849957</td>
<td>-0.07822326652898785</td>
</tr>
<tr>
<td>±3</td>
<td>-0.09127176311424948</td>
<td>-0.01686411844287495</td>
</tr>
<tr>
<td>±4</td>
<td>0</td>
<td>0.02674875741080976</td>
</tr>
</tbody>
</table>

**TABLE 8.15**
Impulse responses of the low- and highpass analysis filters for an irreversible 9-7 wavelet transform.
Chapter 8
Image Compression

**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.)
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.
Chapter 8
Image Compression

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).
Chapter 8
Image Compression

![Diagram illustrating image watermarking system](image-watermarking-system.png)

FIGURE 8.52
A typical image watermarking system:
(a) encoder;
(b) decoder.
Chapter 8
Image Compression

**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.
Chapter 8
Image Compression

**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.