ECE160
Multimedia
Lecture 9: Spring 2011
Image Compression Standards
The JPEG Standard

• JPEG is an image compression standard that was developed by the “Joint Photographic Experts Group”. JPEG was formally accepted as an international standard in 1992.

• JPEG is a lossy image compression method. It employs a transform coding method using the DCT (Discrete Cosine Transform).

• An image is a function of $i$ and $j$ (conventionally $x$ and $y$) in the spatial domain.

• The 2D DCT is used as one step in JPEG in order to yield a frequency response which is a function $F(u,v)$ in the spatial frequency domain, indexed by two integers $u$ and $v$. 
Observations for JPEG Image Compression

• **Observation 1**: Useful image contents change relatively slowly across the image. It is unusual for intensity to vary widely several times in a small area, for example, within an 8x8 pixel block.
  – Much of the information in an image is repeated, hence “spatial redundancy”.

• **Observation 2**: Psychophysical experiments show that humans are less likely to notice the loss of high spatial frequency components than the loss of lower frequency components.
  – The spatial redundancy can be reduced by reducing the high spatial frequency contents.

• **Observation 3**: Visual acuity (accuracy in distinguishing closely spaced lines) is greater for gray (“black and white”) than for color.
  – Chroma subsampling (4:2:0) is used in JPEG.
Main Steps in JPEG Image Compression

- Transform RGB to YIQ or YUV and subsample color.
- DCT on image blocks.
- Quantization.
- Zig-zag ordering and run-length encoding.
- Entropy coding.
Block diagram for JPEG encoder
Block diagram for JPEG encoder
DCT on Image Blocks

• Each image is divided into 8 × 8 blocks. The 2D DCT is applied to each block image \( f(i,j) \), with output being the DCT coefficients \( F(u,v) \) for each block.

• 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.
Block diagram for JPEG encoder

YIQ or YUV

8 × 8

DCT

Quantization

Quantiz. Tables

Coding Tables

Entropy Coding

Header Tables

Data

DPCM

RLC

DC

Zig

Zag

AC
Quantization

\[ f(u, v) = \text{round} \left( \frac{F(u, v)}{Q(u, v)} \right) \]

- \( F(u, v) \) represents a DCT coefficient, 
- \( Q(u, v) \) is a “quantization matrix" entry, and 
- \( F(u, v) \) represents the quantized DCT coefficients which JPEG will use in the succeeding entropy coding.
  - The quantization step is the main source for loss in JPEG compression.
  - The entries of \( Q(u, v) \) tend to have larger values towards the lower right corner. This introduces more loss at the higher spatial frequencies - a practice supported by Observations 1 and 2.
  - The Luminance Quantization Table and The Chrominance Quantization Table provide the default \( Q(u, v) \) values obtained from psychophysical studies to maximize the compression ratio while minimizing perceptual losses in JPEG images.
Luminance Quantization Table

<table>
<thead>
<tr>
<th>16</th>
<th>11</th>
<th>10</th>
<th>16</th>
<th>24</th>
<th>40</th>
<th>51</th>
<th>61</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>12</td>
<td>14</td>
<td>19</td>
<td>26</td>
<td>58</td>
<td>60</td>
<td>55</td>
</tr>
<tr>
<td>14</td>
<td>13</td>
<td>16</td>
<td>24</td>
<td>40</td>
<td>57</td>
<td>69</td>
<td>56</td>
</tr>
<tr>
<td>14</td>
<td>17</td>
<td>22</td>
<td>29</td>
<td>51</td>
<td>87</td>
<td>80</td>
<td>62</td>
</tr>
<tr>
<td>18</td>
<td>22</td>
<td>37</td>
<td>56</td>
<td>68</td>
<td>109</td>
<td>103</td>
<td>77</td>
</tr>
<tr>
<td>24</td>
<td>35</td>
<td>55</td>
<td>64</td>
<td>81</td>
<td>104</td>
<td>113</td>
<td>92</td>
</tr>
<tr>
<td>49</td>
<td>64</td>
<td>78</td>
<td>87</td>
<td>103</td>
<td>121</td>
<td>120</td>
<td>101</td>
</tr>
<tr>
<td>72</td>
<td>92</td>
<td>95</td>
<td>98</td>
<td>112</td>
<td>100</td>
<td>103</td>
<td>99</td>
</tr>
</tbody>
</table>
Chrominance Quantization Table

<table>
<thead>
<tr>
<th>17</th>
<th>18</th>
<th>24</th>
<th>47</th>
<th>99</th>
<th>99</th>
<th>99</th>
<th>99</th>
<th>99</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>21</td>
<td>26</td>
<td>66</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
</tr>
<tr>
<td>24</td>
<td>26</td>
<td>56</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
</tr>
</tbody>
</table>
JPEG compression for a smooth image block

An 8x8 block from the Y image of `Lena'

$$f(i,j)$$

$$F(u,v)$$

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|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|   |   |   |   |   |   |   |   |   |   |   |   |
JPEG compression for a smooth image block

\[
\begin{align*}
F(u, v) & =
\begin{bmatrix}
32 & 6 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\end{align*}
\]

\[
F(u, v) & =
\begin{bmatrix}
512 & 66 & -10 & 0 & 0 & 0 & 0 & 0 \\
-12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-14 & 0 & 16 & 0 & 0 & 0 & 0 & 0 \\
-14 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[
f(i, j) =
\begin{bmatrix}
199 & 196 & 191 & 186 & 182 & 178 & 177 & 176 \\
201 & 199 & 196 & 192 & 188 & 183 & 180 & 178 \\
203 & 203 & 202 & 200 & 195 & 189 & 183 & 180 \\
202 & 203 & 204 & 203 & 198 & 191 & 183 & 179 \\
200 & 201 & 202 & 201 & 196 & 189 & 182 & 177 \\
200 & 200 & 199 & 197 & 192 & 186 & 181 & 177 \\
204 & 202 & 199 & 195 & 190 & 186 & 183 & 181 \\
207 & 204 & 200 & 194 & 190 & 187 & 185 & 184 \\
\end{bmatrix}
\]

\[
f(i, j) =
\begin{bmatrix}
1 & 6 & -2 & 2 & 7 & -3 & -2 & -1 \\
-1 & 4 & 2 & -4 & 1 & -1 & -2 & -3 \\
0 & -3 & -2 & -5 & 5 & -2 & 2 & -5 \\
-2 & -3 & -4 & -3 & -1 & -4 & 4 & 8 \\
0 & 4 & -2 & -1 & -1 & 5 & -2 \\
0 & 0 & 1 & 3 & 8 & 4 & 6 & -2 \\
1 & -2 & 0 & 5 & 1 & 1 & 4 & -6 \\
3 & -4 & 0 & 6 & -2 & -2 & 2 & 2 \\
\end{bmatrix}
\]

\[
\epsilon(i, j) = f(i, j) - f(i, j)
\]
JPEG compression for a textured image block

Another 8x8 block from the Y image of `Lena'

\[
f(i, j) \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \frac{f(i, j)}{F(u, v)}
### JPEG compression for a textured image block

\[
\begin{align*}
F(u, v) &= (-5 -4 9 -5 2 1 1 0) \\
& \quad (-11 -5 -2 0 1 0 0 -1) \\
& \quad (3 -6 4 0 -3 -1 0 1) \\
& \quad (0 1 -1 0 1 0 0 0) \\
& \quad (0 0 -1 0 0 1 0 0) \\
& \quad (0 -1 1 -1 0 0 0 0) \\
& \quad (0 0 0 0 0 0 0 0) \\
& \quad (0 0 0 0 0 0 0 0)
\end{align*}
\]

\[
\begin{align*}
F(u, v) &= (-80 -44 90 -80 48 40 51 0) \\
& \quad (-132 -60 -28 0 26 0 0 -55) \\
& \quad (42 -78 64 0 -120 -57 0 56) \\
& \quad (0 17 -22 0 51 0 0 0) \\
& \quad (0 0 -37 0 0 109 0 0) \\
& \quad (0 -35 55 -64 0 0 0 0) \\
& \quad (0 0 0 0 0 0 0 0) \\
& \quad (0 0 0 0 0 0 0 0)
\end{align*}
\]

\[
\begin{align*}
F(i, j) &= 70 60 106 94 62 103 146 176 \\
& \quad 85 101 85 75 102 127 93 144 \\
& \quad 98 99 92 102 74 98 89 167 \\
& \quad 132 53 111 180 55 70 106 145 \\
& \quad 173 57 114 207 111 89 84 90 \\
& \quad 164 123 131 135 133 92 85 162 \\
& \quad 141 159 169 73 106 101 149 224 \\
& \quad 150 141 195 79 107 147 210 153
\end{align*}
\]

\[
\begin{align*}
\epsilon(i, j) &= f(i, j) - \hat{f}(i, j)
\end{align*}
\]
Block diagram for JPEG encoder
Run-length Coding (RLC) on AC coefficients

- RLC aims to turn the $\hat{F}(u,v)$ values into sets \{#-zeros-to-skip , next non-zero value\}.
- To make a long run of zeros more likely, a zig-zag scan is used on the 8x8 matrix $\hat{F}(u,v)$.
Block diagram for JPEG encoder
DPCM on DC coefficients

• The DC coefficients are coded separately from the AC ones.

• *Differential Pulse Code Modulation (DPCM)* is the coding method.

• If the DC coefficients for the first 5 image blocks are 150, 155, 149, 152, 144, then the DPCM would produce 150, 5, -6, 3, -8, assuming \( d_i = DC_{i+1} - DC_i \), and \( d_0 = DC_0 \).
Block diagram for JPEG encoder
Entropy Coding

- The DC and AC coefficients finally undergo an entropy coding step to gain a possible further compression.
- Use DC as an example: each DPCM coded DC coefficient is represented by (SIZE, AMPLITUDE),
  - SIZE indicates how many bits are used to represent the coefficient,
  - AMPLITUDE contains the actual bits.
- In the example, codes 150, 5, −6, 3, −8 will become
  \((8, 10010110), (3, 101), (3, 001), (2, 11), (4, 0111)\).

  - SIZE is Huffman coded since smaller SIZEs occur more often.
  - AMPLITUDE is not Huffman coded, its value can change widely so Huffman coding has no benefit.
Four Common JPEG Modes

• Sequential Mode - the default JPEG mode, implicitly assumed in the discussions so far. Each graylevel image or color image component is encoded in a single left-to-right, top-to-bottom scan.

• Progressive Mode.

• Hierarchical Mode.

• Lossless Mode - discussed in Chapter 7, to be replaced by JPEG-LS
Progressive Mode

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

1. **Spectral selection**: Uses “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

• 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.
Encoder and Decoder for Three-level Hierarchical JPEG
Encoder and Decoder for Three-level Hierarchical JPEG
Encoder and Decoder for Three-level Hierarchical JPEG
Encoder and Decoder for Three-level Hierarchical JPEG
Encoder for Three-level Hierarchical JPEG

1. Reduction of image resolution: Reduce resolution of the input image \( f \) (e.g., 512x512) by a factor of 2 in each dimension to obtain \( f_2 \) (e.g., 256x256). Repeat this to obtain \( f_4 \) (e.g., 128x128).

2. Compress low-resolution image \( f_4 \): Encode \( f_4 \) using any other JPEG method (e.g., Sequential, Progressive) to obtain \( F_4 \).

3. Compress difference image \( d_2 \):
   (a) Decode \( F_4 \) to obtain \( \tilde{f}_4 \). Use any interpolation method to expand \( \tilde{f}_4 \) to be of the same resolution as \( f_2 \) and call it \( E(\tilde{f}_4) \).
   (b) Encode difference \( d_2 = f_2 - E(\tilde{f}_4) \) using any other JPEG method (e.g., Sequential, Progressive) to generate \( D_2 \).

4. Compress difference image \( d_1 \):
   (a) Decode \( D_2 \) to obtain \( \tilde{d}_2 \); add it to \( E(\tilde{f}_4) \) to get \( \tilde{f}_2 = E(\tilde{f}_4) + \tilde{d}_2 \) which is a version of \( f_2 \) after compression and decompression.
   (b) Encode difference \( d_1 = f - E(\tilde{f}_2) \) using any other JPEG method (e.g., Sequential, Progressive) to generate \( D_1 \).
Decoder for Three-level Hierarchical JPEG

1. Decompress the encoded low-resolution image $F_4$:
   - Decode $F_4$ using the same JPEG method as in the encoder to obtain $\tilde{f}_4$.

2. Restore image $\tilde{f}_2$ at the intermediate resolution:
   - Use $E(\tilde{f}_4) + \tilde{d}_2$ to obtain $\tilde{f}_2$.

3. Restore image $\tilde{f}$ at the original resolution:
   - Use $E(\tilde{f}_2) + \tilde{d}_1$ to obtain $\tilde{f}$. 
The JPEG Bitstream

- Start_of_image
- Frame
- End_of_image

- Tables, etc.
- Header
- Scan
- Scan

- Tables, etc.
- Header
- Segment
- Restart
- Segment
- Restart
- ...

- Block
- Block
- Block
- ...

ECE160
Spring 2011
Lecture 9
Image Compression Standards
The JPEG2000 Standard

• Design Goals:
  – To provide a better rate-distortion tradeoff 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**: There is 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 standard offers excellent rate-distortion performance in mid and high bit-rates. However, at bit-rates below 0.25 bpp, subjective distortion becomes unacceptable. This is important if we hope to receive images on our web-enabled ubiquitous devices, such as web-aware wristwatches and so on.

- **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 standard has 44 modes, many of which are application specific and not used by the majority of JPEG decoders.
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.

• In addition, JPEG2000 is able to handle up to 256 channels of information whereas the current JPEG standard is only able to handle three color channels.
Properties of JPEG2000 Image Compression

- Uses Embedded Block Coding with Optimized Truncation (EBCOT) algorithm which partitions each subband LL, LH, HL, HH produced by the wavelet transform into small blocks called “code blocks”.
- A separate scalable bitstream is generated for each code block => improved error resilience.
Main Steps of JPEG2000 Image Compression

- Embedded Block coding and bitstream generation.
  1. Bitplane coding.
  2. Fractional bitplane coding.
- Post compression rate distortion (PCRD) optimization.
- Layer formation and representation.
Layer Formation and Representation

- JPEG2000 offers both resolution and quality scalability through the use of a layered bitstream organization and a two-tiered coding strategy.
- The first tier produces the embedded block bitstreams while the second tier compresses block summary information.
- Quality layer $Q_1$ contains the initial $R_i^{n_i^1}$ bytes of each code block $B_i$ and the other layers $Q_q$ contain the incremental contribution $L_{q_i} = R_i^{n_i^q} - R_i^{n_i^{q-1}} \geq 0$ from code block $B_i$. 
Layer Formation and Representation

- Three quality layers with eight blocks each.
Region of Interest Coding

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.
Region of Interest Coding

- Region of interest (ROI) coding of an image using a circularly shaped ROI. (a) 0.4 bpp, (b) 0.5 bpp, (c) 0.6 bpp, and (d) 0.7 bpp
Performance comparison for JPEG and JPEG2000

- Natural images.
- Computer generated images.
- Medical images.
Comparison of JPEG and JPEG2000.

Original image.
• (b) JPEG (left) and JPEG2000 (right) images compressed at 0.75 bpp.
• (c) JPEG and JPEG2000 images compressed at 0.25 bpp.
The JPEG-LS Standard

• JPEG-LS is in the current ISO/ITU standard for lossless or “near lossless" compression of continuous tone images.
• It is part of a larger ISO effort aimed at better compression of medical images.
• Uses the LOCO-I (LOw COmplexity LOssless COmpression for Images) algorithm proposed by Hewlett-Packard.
• Motivated by the observation that complexity reduction is often more important than small increases in compression offered by more complex algorithms.
JBIG and JBIG-2: Bi-level Image Compression Standards

Main Goal: Enables the handing of documents in electronic form.

• Primarily used to code scanned images of printed or handwritten text, computer generated text, and facsimile transmissions.

• JBIG is a lossless compression standard. It also offers progressive encoding/decoding capability, the resulting bitstream contains a set of progressively higher resolution images.

• JBIG-2 introduces model-based coding - similar to context-based coding. It supports lossy compressions well.