Chapter 4
Still Image Compression Standards

JPEG and JPEG2000

JPEG
Joint Photographic Experts Group
ISO/IEC JTC1 SC29 WG1
Formed in 1986 by ISO and CCITT (ITU-T)
Became International Standard (IS) in 1991
Compression ratio 10 to 50; 0.5 to 2 bpp. At 1 bpp, one
256×256 image takes only 2 sec at 33.6 kbits/s
Digital Compression and Coding of Continuous-Tone
Still Images (grayscale or color)
Picture Formats

- Up to 65535 lines and 65535 pels/line
- 8 or 12 bits precision
- Color-space independent
  - Up to 255 color components
  - Each component can be subsampled
  - Interleaving
  - To save bits: YUV is better than RGB
- Typical picture sizes

<table>
<thead>
<tr>
<th></th>
<th>CGA ~SIF</th>
<th>VGA ~CCIR601</th>
<th>SVGA ~HDTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pels/line</td>
<td>320</td>
<td>640</td>
<td>~1280</td>
</tr>
<tr>
<td>Lines</td>
<td>240</td>
<td>480</td>
<td>~960</td>
</tr>
</tbody>
</table>

Basic Framework
Linear Transform

- De-correlate pixels
- Allow the most efficient representation
  - Energy concentration
  - Removal or heavy quantization of some coefficients
- Allow perceptually weighted quantization
- Easy for entropy coding
- Karhunen-Loeve Transform (KLT) is optimal
- Discrete Cosine Transform (DCT)
  - Close to KLT for typical images
  - Widely used in JPEG, H.26x, MPEG

2D Discrete Cosine Transform

- DCT

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

Integer DCT by Gonzales is used
DC and AC Coefficients

(Zero-shift to [-128, 127])

DCT Basis Functions
Quantization

• 8x8 quantization table $Q[u,v]$
  – High-freq coefficients can be quantized more
  – Color components can be quantized more

• A scale factor applied to a fixed $Q$

• $Q: \quad FQ_{uv} = \text{round} \left( \frac{C_{uv}}{Q[u,v]} \right)$

• $IQ: \quad C_{uv} = FQ_{uv} \times Q[u,v]$
Example Quantization Tables

• Luminance

<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>105</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

<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>
</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>
</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>
</tr>
<tr>
<td>47</td>
<td>66</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
</tr>
</tbody>
</table>

人眼敏感度 VS 空間頻率

Eye Sensitivity

NIU, Multimedia Communication Lab. 1-11
Zigzag Scan

- Convert 2-D coefficients block to 1-D coefficients
- To generate long runs of zeros

JPEG Encoder

Image block → DCT → Q → DC → DPCM → DC differences

AC → Zigzag → AC coefficients
DC Coding

- **DC Prediction**
  - \( \text{Diff}(n) = \text{DC}(n) - \text{DC}(n-1) \)

- Each \( \text{Diff}(n) \) is coded as
  - Size: in VLC, indicating the size of the following VLI
  - Amplitude: in VLI (variable length integer)

<table>
<thead>
<tr>
<th>Size</th>
<th>Amplitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-1, 1</td>
</tr>
<tr>
<td>2</td>
<td>-3, -2, 2, 3</td>
</tr>
<tr>
<td>3</td>
<td>-7, -6, -5, -4, 4, 5, 6, 7</td>
</tr>
</tbody>
</table>

JPEG Decoder

- DC differences → DPCM → DC
- AC coefficients → Zigzag

Reconstructed image block

NIU, Multimedia Communication Lab.
AC Coding

- Use VLC to code Run-Size
  - Run
- The number of zeros before a nonzero coefficient
  - Size
- The size of the following VLI
- Amplitude
  - The amplitude of the nonzero AC coefficient
  - coded in VLI
- EOB: end-of-block
- ZRL: zero-run-length, a run of 16 zeros

Modes of Operation

- Progressive DCT-based
  - Spectral selection: Low-freq first, high-freq later
  - Successive approximation: MSB first, LSB later
- Mixed Method (P 4-17)
Modes of Operation (cont.)

- Sequential lossless
  - Compression factor only 2 to 3

- Hierarchical
  (Multi-resolution)
  - Pyramid coding (p4-19)

<table>
<thead>
<tr>
<th>Prediction</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a+b-c</td>
<td>a+(b-c)/2</td>
<td>b-(a-c)/2</td>
<td>(a+b)/2</td>
</tr>
</tbody>
</table>

JPEG Picture Quality

- Quality vs. bit rates

<table>
<thead>
<tr>
<th></th>
<th>CGA</th>
<th>VGA</th>
<th>SVGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 bits/pel</td>
<td>poor</td>
<td>fair</td>
<td>good</td>
</tr>
<tr>
<td>1.0 bits/pel</td>
<td>fair</td>
<td>good</td>
<td>excellent</td>
</tr>
<tr>
<td>2.0 bits/pel</td>
<td>good</td>
<td>excellent</td>
<td>excellent+</td>
</tr>
</tbody>
</table>

- Perceptually lossless at 1.5-2 bpp
**Extension to Video**

- Motion JPEG (M-JPEG)
- Compared to MPEG, M-JPEG has
  - No error propagation
  - Random access
  - Low complexity
  - But the compression ratio is low

**Why another standard?**

- In order to address areas that the current standards fail to produce the best quality or performance, as for example:
  - **Low bit-rate compression**: Current standards, such as JPEG, offer excellent rate-distortion performance in the mid and high bit-rates. However, at low bit-rates (e.g., below 0.25 bpp for highly detailed gray-level images) the subject distortion become unacceptable.
  - **Lossless and lossy compression**: There is currently no standard that can provide superior lossless compression and lossy compression in a single codestream.
Why another standard? (cont’d)

- **Computer generated imagery**: The current standard was optimized for natural imagery and does not perform well on computer generated imagery.

- **Compound documents**: Currently, JPEG is seldom used in the compression of compound documents because of its poor performance when applied to bi-level (text) imagery.

- **Transmission in noisy environments**: The current JPEG standard has provision of restart intervals, but image quality suffers dramatically when bit errors are encountered.

- **Large images**: The JPEG compression algorithm does not allow for images greater than 64K by 64K without tiling.

---

JPEG2000 targets

Thus, the aim of the JPEG2000 is to develop a new still image coding standard for different types of still images (bi-level, gray-level, color, multicomponent, hypercomponent), with different characteristics (natural, scientific, medical, remote sensing, text, rendered graphics, compound, etc.), allowing different imaging models (client/server, real-time transmission, image library archival, limited buffer and bandwidth resources, etc.) preferably within a unified and integrated system.

This coding system is intended for low bit-rate applications, exhibiting rate-distortion and subjective image quality performance superior to existing standards.
JPEG2000 Features

- Superior low bit-rate performance
- Lossless and lossy compression
- Progressive transmission by SNR and resolution
- Static and dynamic Region-of-Interest
- Random stream access and processing
- Error resilience
- ✗ Only define DECODER

JPEG2000 vs JPEG baseline

![Graph comparing JPEG2000 and JPEG]
JPEG2000 vs JPEG baseline

Hotel

PSNR [dB]

Bits per pixel [bpp]

Gold

PSNR [dB]

Bits per pixel [bpp]
JPEG2000 vs JPEG baseline

Example 1: Lena coded at bit-rate
0.134 bpp by JPEG
0.131 bpp by JPEG2000

Example 2: Pepper coded at bit-rate
0.1288 bpp by JPEG
0.0983 bpp by JPEG2000
JPEG2000 vs JPEG baseline

Example 3: F16 coded at bit-rate

0.1256 bpp by JPEG
0.09754 bpp by JPEG2000

JPEG2000 vs JPEG baseline

**JPEG at 0.125 bpp**
JPEG2000 vs JPEG baseline

**JPEG2000 at 0.125 bpp**

![Image of a scene with multiple houses and paths]

**JPEG2000 vs JPEG baseline**

**JPEG at 0.25 bpp**

![Image of the same scene in grayscale]

NIU, Multimedia Communication Lab.
JPEG2000 vs JPEG baseline

JPEG2000 at 0.25 bpp

Low Bit-rate Performance (Cont'd)

JPEG compressed at 130:1  JPEG2000 compressed at 187:1
The SNR Scalability of JPEG2000

Resolution Scalability of JPEG2000
Region Of Interest

- 0.0625 bpp

Region Of Interest (cont’d)

- 0.125 bpp
Basic JPEG2000 CODEC

- **Encoder**
  - Source Image Data
  - Wavelet Transform (SSWT)
  - Scalhar quantizer/or trellis coded quantization
  - Block Coding
  - Rate Control and Bit Stream Composition
  - Entropy Coder
  - Bit Stream

- **Decoder:** Reverse the order

---

Stage 1: Wavelet Transform

- JPEG2000 supports various **Wavelet kernels** and allows various **Wavelet decompositions**.

- For low memory constraint, it uses **Spatially Segmented Wavelet Transform (SSWT)**, and the results of SSWT is the same as traditional wavelet transform.
Spatially Segmented Wavelet Transform

- SSWT partition the input image into overlapping tiles.
- The size of the overlapped part is proportional to the taps of selected wavelet kernels.
- Each tile is processed independently.

An Example for SSWT

Original image → tiles → subband samples
Stage 2: Quantization

- It implements two quantization methods:
  - Scalar Quantization (Part I), and
  - Trellis Coded Quantization (Part II).

Quantizer: Scalar Quantization

- Memoryless

\[ q_b[n] = \text{sgn}(s_b[n]) \times \left\lfloor \frac{s_b[n]}{\Delta_b} \right\rfloor \]

- \( q_b[n] \): the quantized sample indices
- \( s_b[n] \): the unquantized sample values
- \( \Delta_b \): the quantization step size
Quantizer: Scalar Quantization

- Embedded quantizer

\[ q_b^p[n] = \text{sgn}(s_b[n]) \left\lceil \frac{s_b[n]}{\Delta_b^p} \right\rceil \]

where \( \Delta_b^p = 2^p \Delta_b \)

\( q_b^p[n] \) can be obtained by ignoring \( p \) LSB bits from \( q_b[n] \)

Stage 3: Entropy Encoder

- Contains two process:
  - Block Coding.
    - Types of coding operation
      - Zero coding (ZC)
      - Run-Length coding (RLC)
      - Sign coding (SC)
      - Magnitude refinement (MR)
      - Arithmetic coding is used
  - Embedded Block Coding with Optimized Truncation (EBCOT)
**Process 1: Block Coding**

- Each subband is partitioned into a set of blocks (say $32 \times 32$ or $64 \times 64$)
- Blocks are encoded by arithmetic coding engine independently
- The probability models used by the arithmetic coder evolve different combinations of ZC, RLC, SC, and MR.

**An Example for Block Coding**

For each block, $B_i$, a separate bit-stream is generated without information from the other blocks.
An Example for Block Coding

- Code-blocks and packet partition location

Process 2: EBCOT

- Embedded Block Coding with Optimized Truncation.
  - The bit-stream can be truncated to a variety of discrete length, $R_1^1, R_1^2, R_1^3 \ldots$ at point $n=1, 2, 3, \ldots$.
  - The distortion incurred when reconstructing from each of these truncated subset is $D_1^1, D_1^2, D_1^3 \ldots$.
  - Determines the point that each block's bit-stream should be truncated (all sub-bitplanes with rate-distortion slope $\geq th$ being collected together).
  - Rate-distortion optimization is performed but it does not need to be standardized.
Conclusions

- Better than transform coding, but with higher computation load.
- Not supposed to replace JPEG
- Better than subband coding.
- MJPEG2000 (Part III) or MPEG ?
- MPEG-4 : EZW for sprite coding
  - arbitrary shape