JPEG-2000

- Joint Photographic Experts Group ISO/IEC JTC1/SC29/WG1
- Still image compression standard
- Features
  - Improved compression efficiency (vs. JPEG)
  - Highly scalable embedded data streams
  - Progressive lossy to lossless compression within a single data stream
  - Arbitrarily crop images in the compressed domain
  - Selectively enhance quality of spatial “regions of interest”
  - Support for very large images
- JPEG-2000 Part I (minimum compliant decoder) international standard since December 2000.
JPEG 2000 compression

1. **Optional tiling**
   - Image component
   - Subdivision of images into rectangles for independent coding

2. **DWT**
   - Reversible 5/3 or floating-point 9/7 Daubechies wavelet.

3. **Embedded deadzone quantization**
   - Uniform scalar quantization, adjustable deadzone

4. **Bitplane coding**

5. **R-D optimal code-block truncation**
   - Embedded block coding with optimal truncation (EBCOT)

bits
Partitioning of image subbands into code-blocks

Embedded code-block bitstreams

Embedded code-block bitstreams
Optimal bit-stream truncation

- Find optimal truncation points $z_i$ through Lagrangian formulation

\[
J(z_0, z_1, z_2, \ldots) = \sum_{\text{blocks } i} J_i(z_i) = \sum_{\text{blocks } i} D_i(z_i) + \lambda R_i(z_i) \rightarrow \min.
\]

- “Equal slope” condition for distortion-rate functions $D_i(R_i)$

- Feasible truncation points lie on convex hull

![Graph showing the relationship between $D_i(R_i)$ and $R_i$.]
Embedded deadzone uniform quantizers

Supported in JPEG-2000 with general $\beta$ for quantization of wavelet coefficients.
Bit-plane coding for embedded quantization

-2 11 0 -23 49 3 -10

Sign bits
Magnitude bit planes

↑ MSB
↓ LSB

Zero magnitude or positive sign bit
Non-zero magnitude or negative sign bit
Conditional coding of bit-planes

- Encoding takes already encoded, more significant bit-planes into account
- “Significant” wavelet coefficient: magnitude was outside of deadzone in a previous encoded bit-plane

Three “primitive” coding operations

- **Significance coding**: coefficient not yet significant. Arithmetic coding with 9 coding contexts, based on significance of neighbors in same band. Run mode.
- **Sign coding**: coefficient just changed from insignificant to significant. Arithmetic coding with 5 coding contexts.
- **Magnitude refinement coding**: coefficient is already significant. Arithmetic coding with 3 coding contexts.

- Adaptive coding: must maintain 18 probability models
Significance/sign/magnitude refinement coding

Sign bits
Magnitude bit planes

Significance coding
Magnitude refinement coding
Sign coding
Stripe-oriented scanning pattern

Stripe

Context window
Fractional bit-planes

- Three passes for each bit-plane
  - Significance Propagation Pass
  - Magnitude Refinement Pass
  - Cleanup Pass
- Generate additional feasible truncation points for each block

\[ D_i(R_i) \]
Fractional bit-planes (cont.)

1. Significance Propagation Pass
   - Only encode coefficients
     - that are insignificant
     - possess significant neighbors (within 8-neighborhood, including those just became significant in this pass)

2. Magnitude Refinement Pass
   - Encode all coefficient that were already significant in the previous bit-plane

3. Cleanup Pass
   - Encode all remaining (insignificant) coefficients
JPEG2000 compression results

Original Carol Image (512 x 512 Pixels, 24-Bit RGB, Size 786K)
JPEG2000 compression results

75:1, 10.6 Kbyte
JPEG2000 compression results

150:1, 5.3 Kbyte
JPEG2000 compression results

300:1, 2.6 Kbyte
Comparison JPEG vs. JPEG2000

Lenna, 256x256 RGB
Baseline JPEG: 4572 bytes

Lenna, 256x256 RGB
JPEG-2000: 4572 bytes
Comparison JPEG vs. JPEG2000

JPEG with optimized Huffman tables
8268 bytes

JPEG2000
8192 bytes
Reading

- Taubman, Marcellin, Chapter 8, up to section 8.3.3