JPEG2000 vs. MPEG−4 Natural Still Image Coding

Peter Meerwald, pmeerw@cosy.sbg.ac.at

Common requirements & Scalability
JPEG2000 Overview
EBCOT Coding
MPEG−4 Image Coding Overview
MZTE Overview
MZTE, ZTE & EZW Coding

Discussion & Remarks

Common requirements

- compression efficiency
- arbitrary shape / region−of−interest (ROI) coding
- spatial scalability
- quality (SNR) scalability

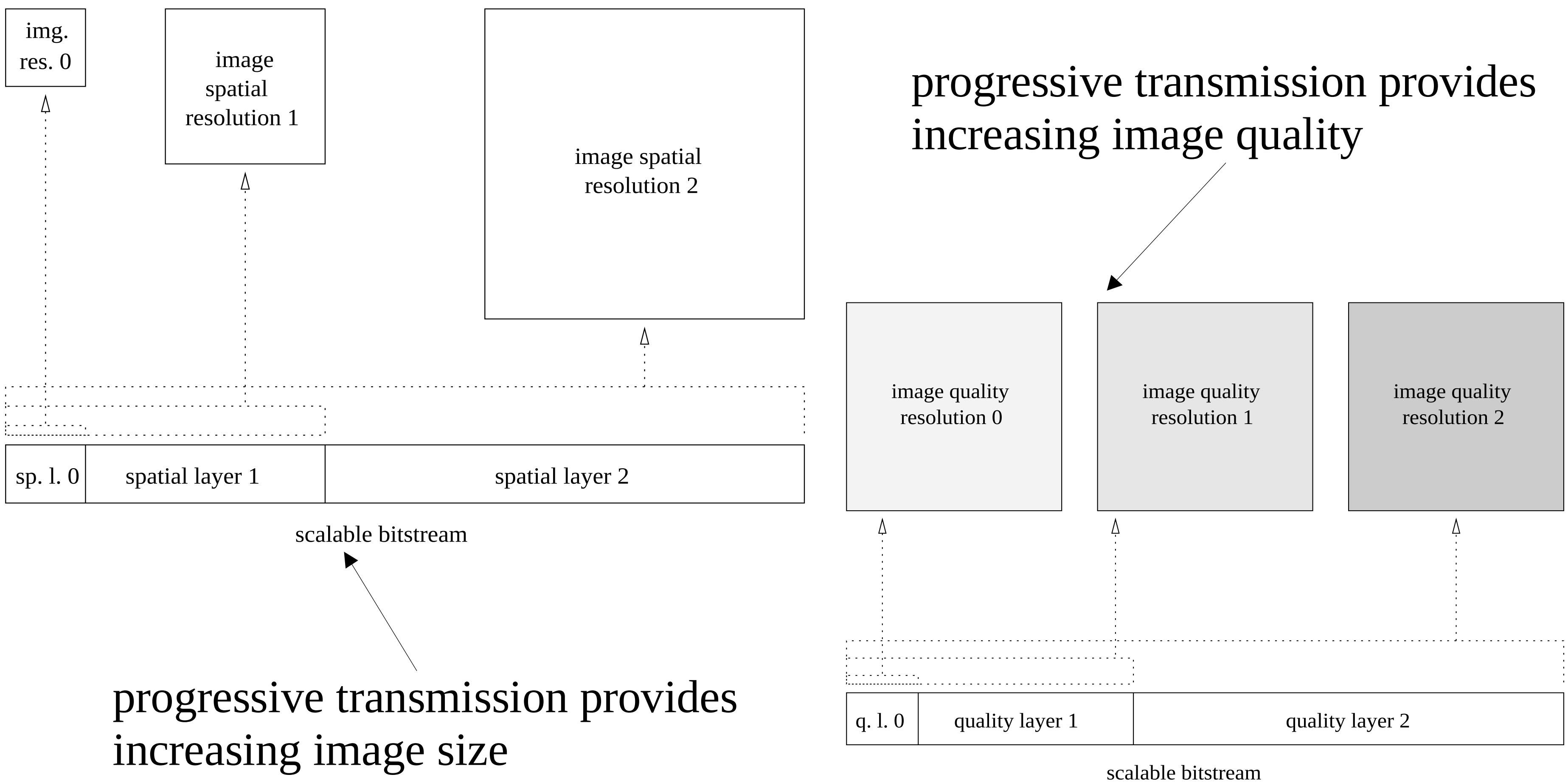
other features:

- random access,
- error−resiliance,
- low complexity,
- memory efficiency

Wavelet transform has interesting properties to achieve goals [1].

[1] Z. Xiong et al., A comp. study of DCT− and Wavelet−based image coding, IEEE Trans. Circ. Sys. Vid.Tech., v. 9, n. 5, August 1999.

Spatial vs. Quality Scalability



Implementations

- JPEG2000 ISO/IEC 15444 (December 2000)
  - jj2000 4.1 (EPFL), http://jj2000.epfl.ch
  - Jasper 1.2 (by. M. D. Adams), http://www.ece.ubc.ca/~mdadams/jasper/
  - Kakadu 2.0 (by D. Taubman), http://maestro.ee.unsw.edu.au/~taubman/kakadu/
  - VM 8.6 (not public)
- MPEG4 ISO/IEC 14496 (August 1999)
  - European ACTS project MoMuSys
  - Microsoft, both via http://www.iso.ch

JPEG2000 Overview [4,5]

- lossy and lossless operation
- gray−scale and multi−component images
- features: scalability, ROI, random access, error resiliance, low complexity, good image quality
- free of royalties
- standard defines decoder, allows for extentions

ISO/IEC 15444 Parts

1 − core coding system; minimum functionality
2 − extensions; additional functionality
3 − Motion JPEG2000; intra−frame video coding
4 − conformance testing
5 − reference software

[4] M. Marcellin et al., An Overview of JPEG−2000, Proc. IEEE Data Comp. Conf., pp. 523 − 541, March 2000.
[5] M. D. Adams et al., JPEG2000: The Next Generation Still Image Compression Standard, ISO/IEC JTC 1/SC 29/WG 1 N 1734, June 2000.

JPEG2000 Terminology (1)

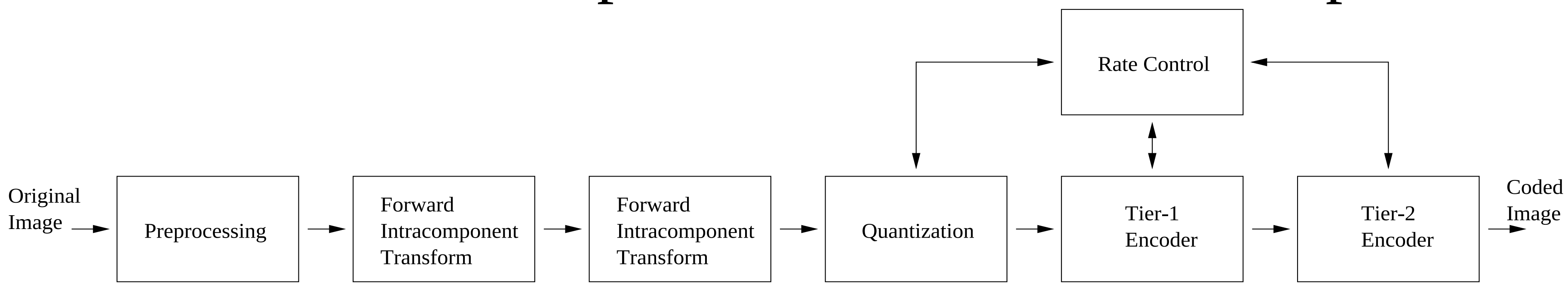
- image
  - one or more *components* (colors, sub−sampling)
  - reference grid (to fix geometry of components)
  - *tiling* (for memory efficiency), *tile−component*
- *code−blocks*
  - partitioning of a subband
  - width and height power−of−two, size maximum 4096
- *coding pass*
  - associated with component, level, subband, code−block

JPEG2000 Terminology (2)

- *packet*
  - collection of coding pass data: header and body
  - generated for each component, level, *layer*, and *precinct*
- *layers*
  - for scalability, rate control must decide in which layer to include coding pass
- *precinct*
  - grouping of code−blocks within a subband
  - coded in separate packets, for error resiliance

EBCOT − Embedded Block Coding with Optimized Truncation [6]

- wavelet/subband coding
- bit−plane coding, multi−pass
- context−adaptive arithmetic coder, MQ−coder [7]
- rate/distortion optimization via truncation points



[6] D. Taubman, High Performance Scalable Image Compression with EBCOT, IEEE Trans. Image Proc., v. 9, n. 7, pp. 158 − 1170, July 2000.
[7] ISO/IEC 14492, Lossy/lossless coding of bi−level images, JBIG2 standard, 2000.

Preprocessing and Transforms

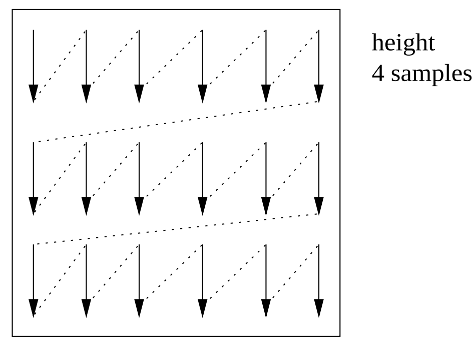
- preprocessing: input data has to be symmetrically distributed about zero
- intercomponent transform (RGB to YCrCb)
  - ICT: Irreversible Color Transform
  - RCT: Reversible Color Transform
- Intracomponent Transform [8] (DWT)
  - reversible integer−to−integer (5/3)
  - nonreversible real−to−real (9/7)
  - lifting, symmetric extension

[8] M. D. Adams et al., Reversible Integer−to−Integer Wavelet Transforms for Image Compression, IEEE Trans. Image Proc. v. 9, n. 6, pp. 1010 − 1024, June 2000.

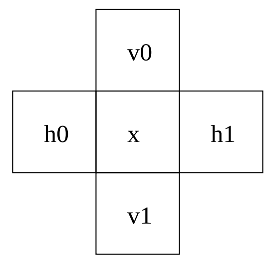
Tier−1: Bit−plane Coding

quantized indices of each subband are partitioned into code−blocks and independently coded (no interband dependencies) in 3 passes

- Significance Pass
  - significance and sign information
  - prediction from previous bit−plane, 9 contexts

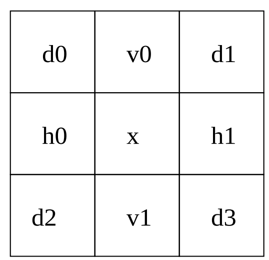


- Refinement Pass
  - code next bit of significant sample, 3 contexts



- Cleanup Pass
  - code not yet handled and mis−predicted samples
  - vertical aggregation mode

→ context selection
→ lazy mode
→ termination mode



Tier−2: Packetization

- one packet for each
  - component
  - level
  - layer
  - precint

➔ other orderings possible
➔ interleaving of tiles
➔ ...

- 5 builtin progressions, packet ordering in code stream
  - layer−resolution−component−position
  - resolution−layer−component−position
  - resolution−position−component−layer
  - position−component−resolution−layer
  - component−position−resolution−layer

Rate Control (1)

- two mechanisms
  - quantizer step size ← can be employed for ROI coding or perceptual weighting − inefficient for rate control
  - selection of coding passes
- encoder knows the contribution each coding pass makes to rate, R, and distortion, D (metrics: MSE, perceptually weighted...)
- have list of (rate, distortion) pairs for each code−block i, each pair is a possible *truncation point* n_i
- any set of truncation points $\{n_i^\lambda\}$ which minimizes

$$D(\lambda)+\lambda R(\lambda)=\sum_i \left(D_i^{n_i^\lambda}+\lambda R_i^{n_i^\lambda}\right)$$

for some λ is optimal; need to find value of λ such that $R(\lambda)=R_{max}$

- perform minimization of $D_i^{n_i^\lambda}+\lambda R_i^{n_i^\lambda}$ for each code−block
- need to try for many different λ using binary search (have good bounds)

Rate Control (2)

- algorithm to minimize $D_i^{n_i^\lambda} + \lambda R_i^{n_i^\lambda}$

not in the standard, EBCOT coding, can use any or no optimization

- initialize $n_i^\lambda = 0$
- for j = 1, 2, 3, ...

  set $\Delta R_i^j = R_i^j - R_i^{n_i^\lambda}$ and $\Delta D_i^j = D_i^j - D_i^{n_i^\lambda}$

  if $\Delta D_i^j / \Delta R_i^j > \lambda$ then update $n_i^\lambda = j$

slopes $S_i^{j_k} = \Delta D_i^j / \Delta R_i^j$ must be strictly decreasing, therefore can simply optimal select truncation point n_i^λ for a given λ

$$n_i^\lambda = max\{ j_k \in N_i | S_i^{j_k} > \lambda \}$$

and can easily determine set of feasible truncation points N_i

MPEG−4 Image Coding Overview

- one of many coding tools in MPEG−4, for still image and texture coding, wavelet−based (9/3)
- MZTE coding with three quantization modes [9]
  - single quantization
  - multi quantization
  - bilevel quantization (PEZW?)
- extension for arbitrary shape coding [10,11,12]
- was JPEG2000 candidate

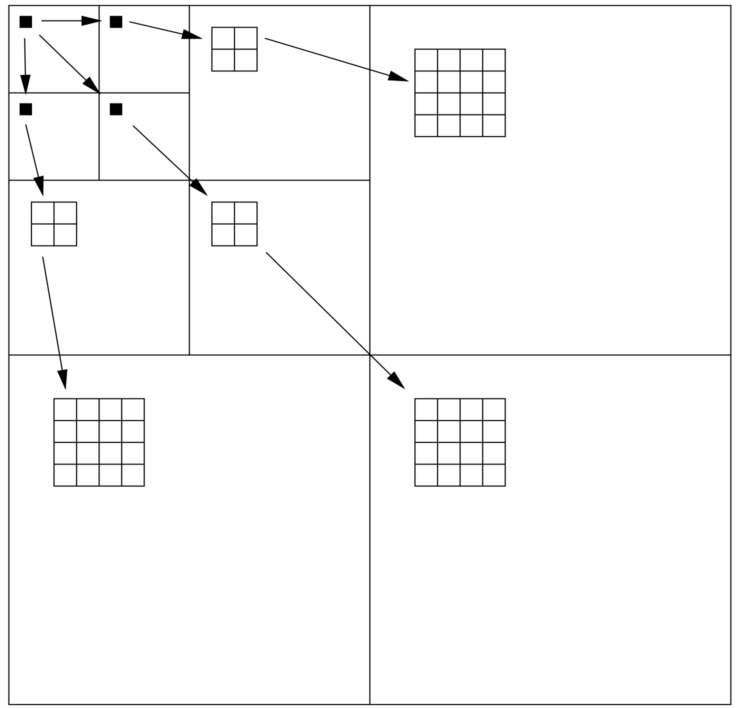
developed by Sarnoff Corp.

TI?

Microsoft

[9] I. Sodagar et al., Scalable Wavelet Coding for Synthetic/Natural Hybrid Images, IEEE Trans. Circ. Sys. Video. Tech., v. 9, n. 2, March 1999.
[10] S. Li et al., Shape−adaptive wavelet coding, Proc. IEEE Symp. Circ. Sys., v. 5, pp. 281 − 284, May 1998.
[11] G. Xing et al., Arbitrarily shaped video object coding by wavelet, Proc. IEEE Symp. Circ. Sys., May 2000.
[12] S. Li et al., Generic, Scalable and Efficient Shape Coding for Visual Texture Objects in MPEG−4, Proc. IEEE Symp. Circ. Sys., May 2000.

Embedded Zerotree Wavelet (EZW) Coding



- exploiting hierarchical self−similarity through Zerotree hypothesis [13]
- encoding coeff. significance using four symbols:

  NS − *negative significant*

  PS − *positive significant*

  ZTR − *zerotree root, children insignificant*

  IZ − *isolated zero, child significant*

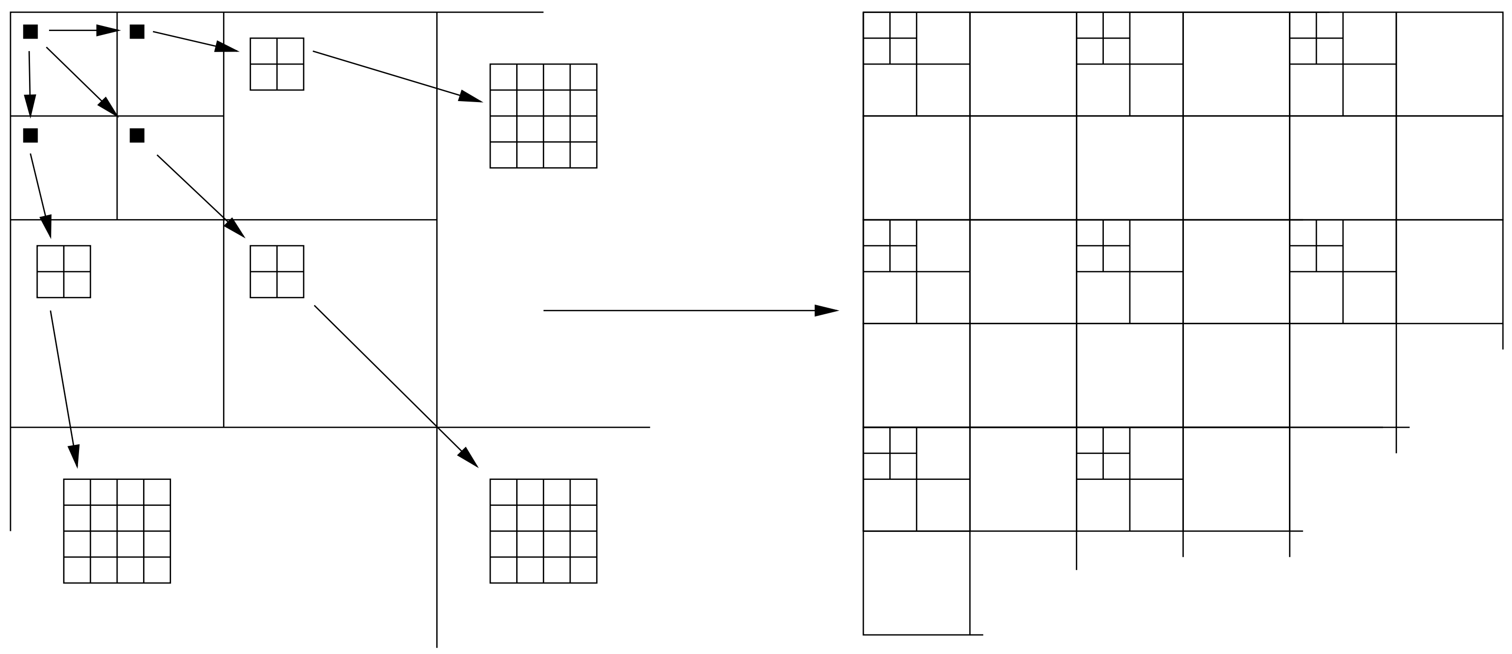
- encoding significant coeff. using successive approx. quant. (SAQ), bit−plane coding [14]

„If a wavelet coefficient at a coarse scale is *insignificant* with respect to a threshold *T*, then all wavelet coefficients of the *same orientation* at the *same spatial location* at finer wavelet scales are likely to be insignificant as well.“

[13] A. Lewis et al., Image compression using the 2D wavelet transform, IEEE Trans. Image Proc., v. 1 pp. 244 − 250, April 1992.
[14] J. Shapiro, Embedded image coding using zerotrees of wavelet coefficients, IEEE Trans. Signal Proc., v. 41, n. 12, pp. 3445 − 3462, Dec. 1993.

Zerotree Entropy (ZTE) Coding

- based on EZW coding (zerotree hypothesis)
- coefficients re−organized to form a *wavelet block*
- explicit quantization, adaptivity possible
- coefficient scanning: depth−firth or subband−by−subband
- separate coding of approximation and detail subband coefficients



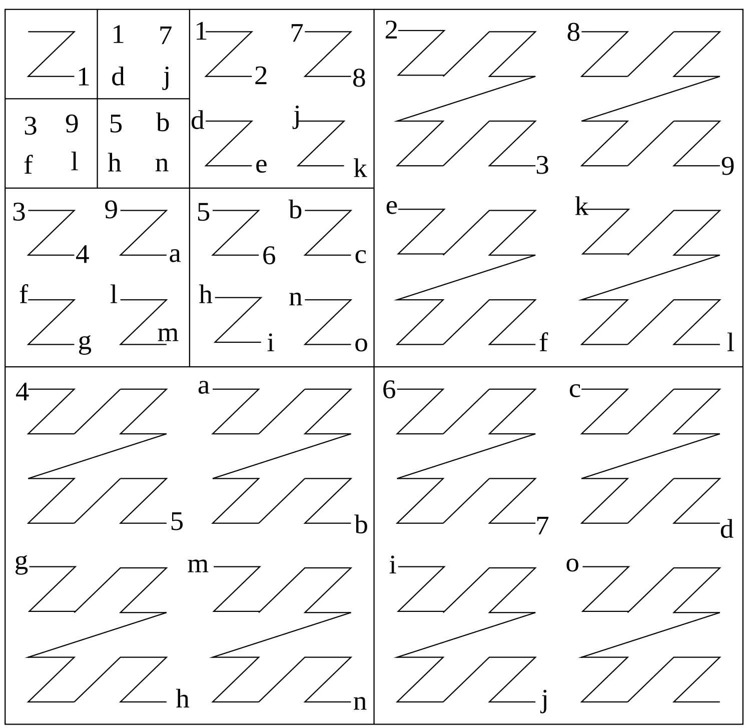
ZTE Symbols and Scan Order

ZTR − zerotree root; insignificant, as well as children

VZTR − valued zerotree root; significant, but all children insignificant

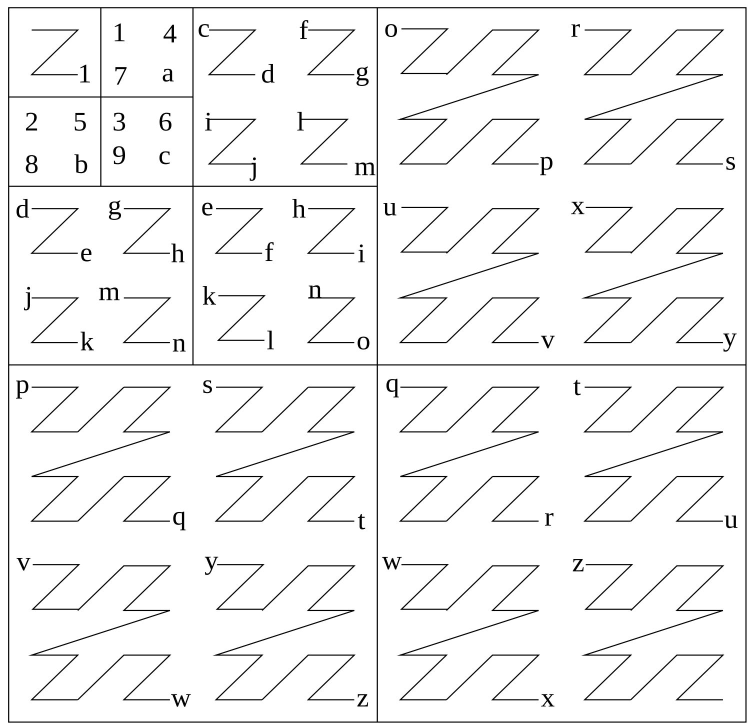
VAL − value, has non−zero descendants

IZ − isolated zero; insignificant, but has non−zero children



tree−depth first scan order

usesless memory,
requires buffer
for progressive
transmission

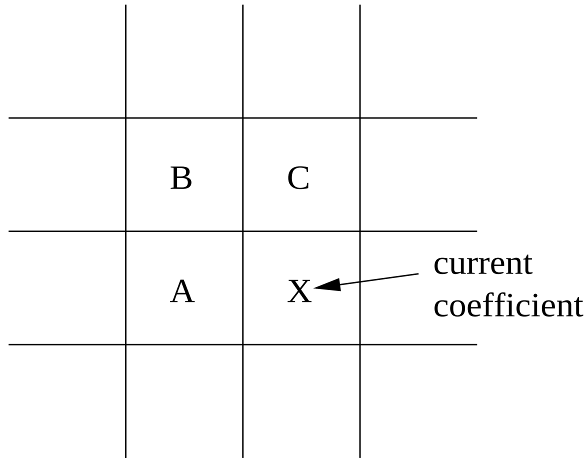


subband−by−subband scan order

ZTE Approx. Coeff. Coding

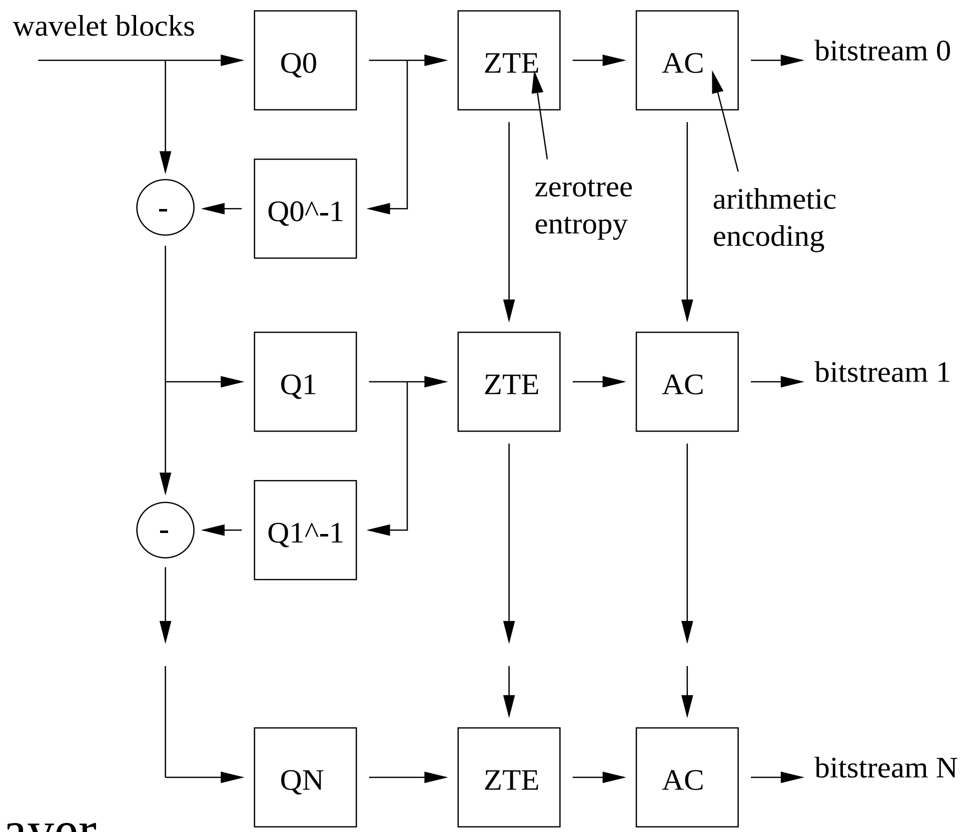
1. quantize coefficients
2. calculate predicted minimum (Dmin) difference and maximum coeff. value (Vmax) after prediction
3. emit quantization step size, Dmin and Vmax
4. subtract Dmin from all predicted differences: Y = X − Dmin
5. encode all Y in raster−scan order

```
forward prediction:
  if |A − B| < |B − C|
    P = C
  else
    P = A
  X = X − P
```



Multiscale Zerotree Wavelet Entropy (MZTE) Coding

- based on ZTE, but improves scalability
- approx. band coded separately



- produce layer 0 using Q0
- compute residual after quantization (Q0)
- encode residual with finer Q1
- leads to a sequence of refinement−Qs

- can re−use sign information of previous layer
- can benefit from two probability models

    case 1: coefficient quantized to zero in previous layer

    case 2: otherwise

Comparing EZW and MZTE

- EZW
  - 4 symbols for significance map (ZTR,IZ,NS,PS)
  - 2 symbols for SAQ
- MZTE
  - 4 symbols for node type (ZTR, VZTR, VAL, IZ)
  - 2 refinement models: one for VZTR/VAL, one for VAL only
  - context modeling between scalability layers, residual model

Shape−adaptive ZTE

- good description in [15]
  - explicit representation of shape (MPEG−4 shape coding tool, developped by OKI) [18]
  - shape encoding uses context−based arithmetic encoding
  - extend zerotrees with ’out−node’, do not code out−nodes
- choice of filter matters (~1 dB) 9/3, D−4, 12/4
- subsampling strategy mattters (~2 dB)

[15] S. Li et al., Shape Adaptive DWTs for Arbitrarily Shaped Visual Object Coding, IEEE Trans. Circ. Sys. Video Tech., August 2000.
[18] C. Jordan et al., Shape Representation and Coding of Visual Objects in MM App., Ann. Telecomm., v. 53, n. 5, pp. 164 − 178, May 1998.

Remarks on Shape−adaptive Coding

- shape−adaptive DWT superior to DCT−based methods [11]
  - intraframe: low−pass extrapolation DCT, ΔDC−SA−DCT
  - interframe: zero padding DCT, SA−DCT
- alignment of the filter important, two strategies [15]
  - even subsampling favoring zerotree coding
  - odd subsampling favoring signal processing gain
- zerotree assumption frequently broken on object boundaries and for the prediction residue; JPEG2000−like coding superior [11]

Remarks on Zerotree Coding Gain

- correlation coefficient between parent and child magnitude ~ 0.35 − motivation for zerotree coding [14]
- experiment: rotate subbands 90° with respect to previous scale (destroys parent−child relationship) [16]
- compare SPIHT performance
  - with and without arithmetic coding, rotation to determine Gpc (increase in dB due to exploiting parent−child relationship)
  - Gpc = 0.25 dB for SPIHT−AC, 0.40 dB for SPIHT−NC

[16] M. Marcellin et al., Quantifying the parent−child coding gain in zerotree based coders, IEEE Signal Proc. Letters, v. 8, n. 3, March 2001.

Remarks on Mutual Subband Information

- compute mutual information [17] $I(X;Y)=E_{X,Y}\log\frac{p(x,y)}{p(x)\dot{p}(y)}=h(X)-h(X\vee Y)$
  - I(X;PX) parent−child relationship
  - I(X;NX) coefficient−neighborhood relationship
  - I(X;PX,NX) combined model
- experimental results
  - I(X;PX,NX) always much larger than I(X;PX)
  - I(X;NX) close to I(X;PX,NX)
  - longer filters (Daubechies−4 vs. 8) favour I(X;NX)

[17] J. Liu et al., An analysis of interscale and intrascale dependencies between image wavelet coefficients, Proc. ICIP, pp. 669 − 672, Sept. 2000.

Summary

- two wavelet−based standards for still image coding with support for scalability and ROI/shape−adaptive coding
- JPEG2000 is newer and seems to be better designed
- extensions for JPEG2000 interesting: perceptual coding; different quantization, decomposition strategies, ...
- performance evaluation in terms of image quality and runtime necessary (taking ’features’ into account)