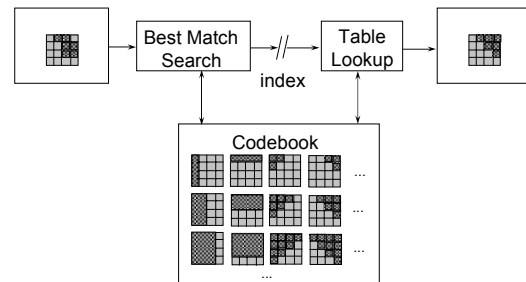


11. Other Compression Techniques

Prof. Chia-Wen Lin
Department of CS
National Chung Cheng University
886-5-272-0411 ext. 33120
cwlin@cs.ccu.edu.tw

Vector Quantization (Cont.)

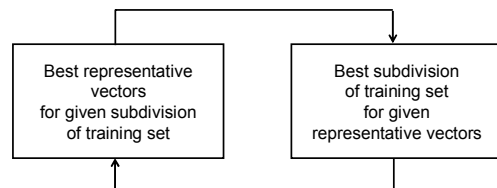


Non-Standard Coding Techniques

- Vector Quantization (VQ)
- Fractal coding
- Subband and Wavelet coding
- Model-based coding

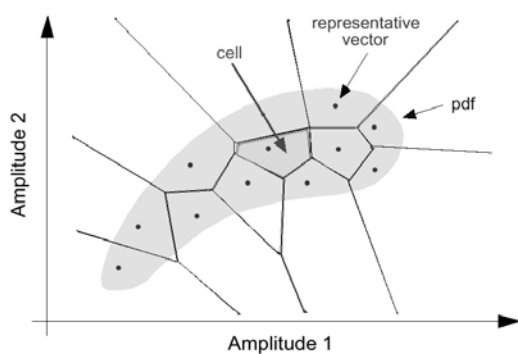
LBG Algorithm

- Linde, Buzo, Gray, 1980: Lloyd algorithm generalized for VQ



- Assumption: fixed codeword length
- Code book unstructured: full search

Vector Quantization



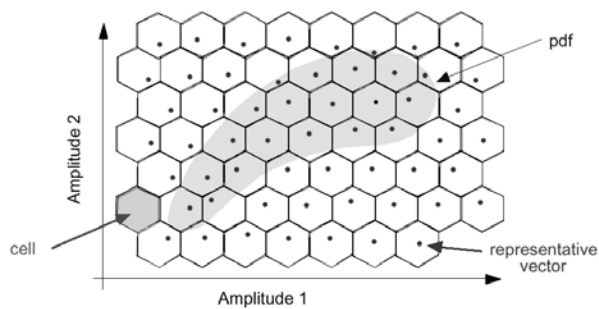
VLC VQ

- Chou, Lookabaugh, Gray, 1989: extended LBG algorithm for entropy-coded VQ
- Lagrangian cost function: solve unconstrained problem rather than constrained problem

$$\min\{D+\lambda R\}$$
- Unstructured codebook: full search for $\min\{D+\lambda R\}$

The most general coder structure:
Any source coder can be interpreted as VQ with VLC

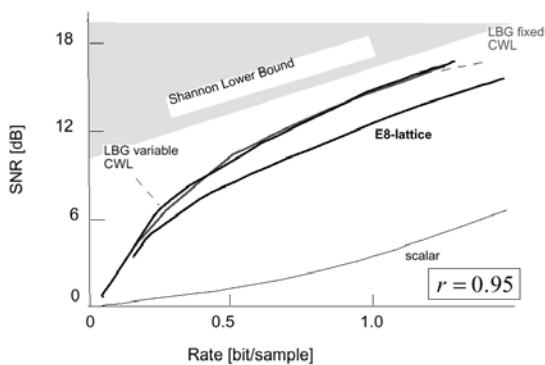
Lattice Vector Quantization



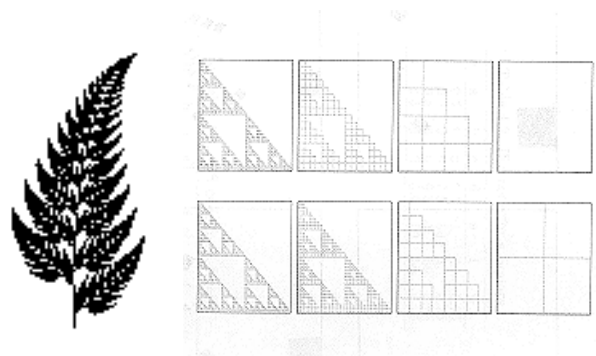
Vector Quantization

- Highly asymmetrical encoding and decoding complexity
- Encoding complexity depends on the codebook size
- Decoding is done simply by table lookup
- Codebook design usually is an iterative process which involves off-line training
- Different codebook designs result in different quality

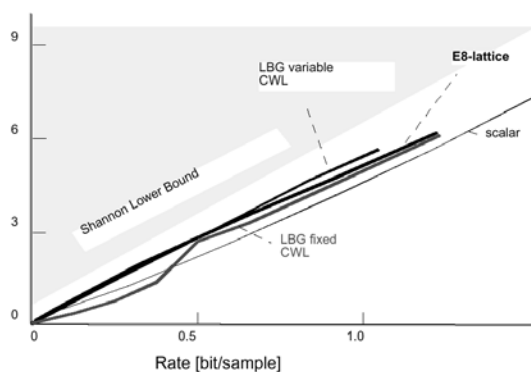
8D VQ of Gauss-Markov Source



Fractals



8D VQ of Memoryless Laplacian Source



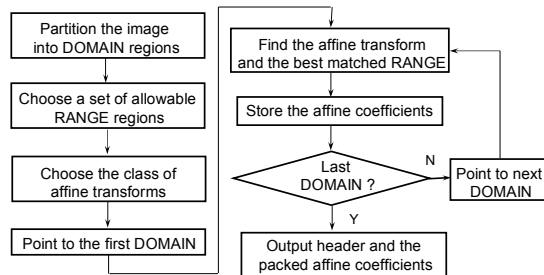
Affine Transformation

Maps points into new points according to the transform:

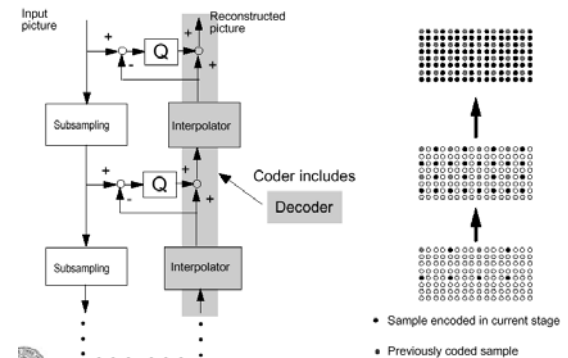
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

- Can model scaling, translation, rotation, etc.

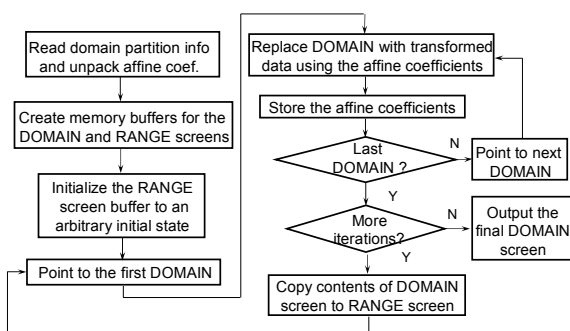
Fractal Image Coding



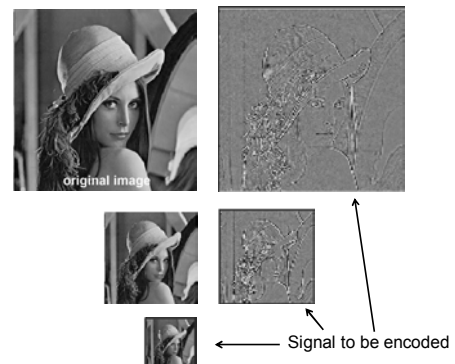
Interpolation Error Coding



Fractal Image Decoding



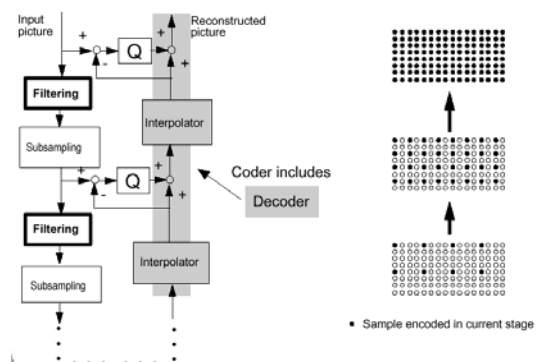
Interpolation Error Coding (Cont.)



Fractal Coding

- Real-time encoding is a problem
- Currently quality may not be better than JPEG and MPEG (especially for high quality applications)
- + Real-time decoding is simple, can be implemented using regular PC
- + Resolution independent
- + Relatively new, may have more room for improvement

Predictive Pyramid Coding



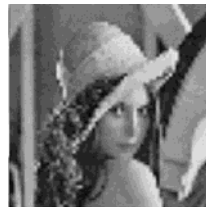
Predictive Pyramid Coding (Cont.)

←
subsampling factor

Interpolation Error Coding vs. Pyramid (Cont.)

Resolution layer #1, interpolated to original size for display

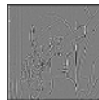
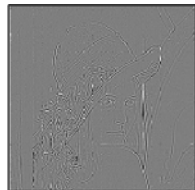
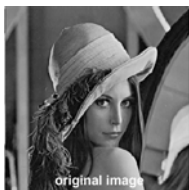
Interpolation error coding



Pyramid



Predictive Pyramid Coding (Cont.)



← Signal to be encoded

Interpolation Error Coding vs. Pyramid (Cont.)

Resolution layer #2, interpolated to original size for display

Interpolation error coding



Pyramid



Interpolation Error Coding vs. Pyramid

Resolution layer #0, interpolated to original size for display

Interpolation error coding



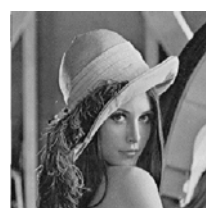
Pyramid



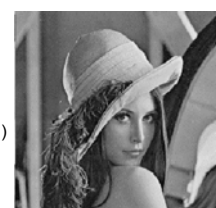
Interpolation Error Coding vs. Pyramid (Cont.)

Resolution layer #3

Interpolation error coding

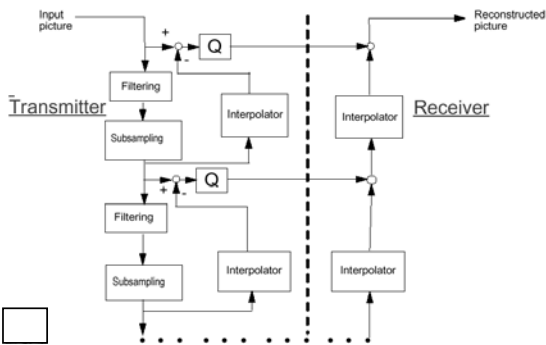


Pyramid

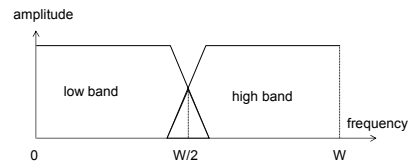


=
(original)

Open-loop Pyramid (Laplacian Pyramid)

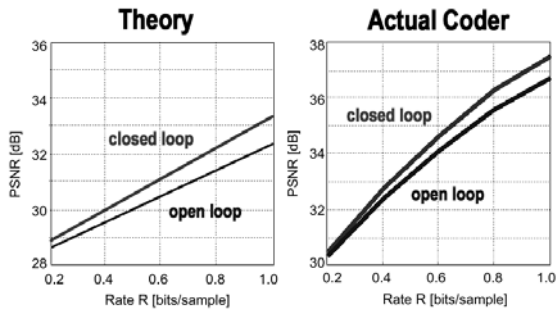


Two-Band Analysis Filter Band

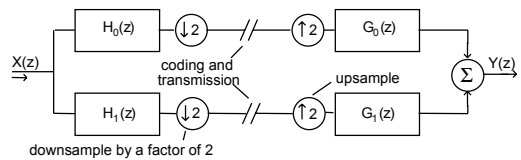


- To eliminate aliasing distortion, the synthesis and analysis filters must have certain relationships

Two-layer Open- vs. Closed-loop Pyramid

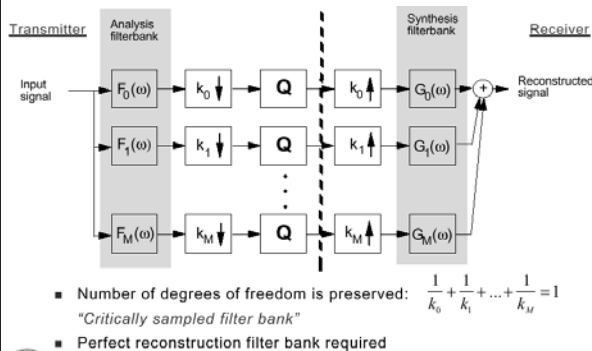


Two-Band Subband Codec

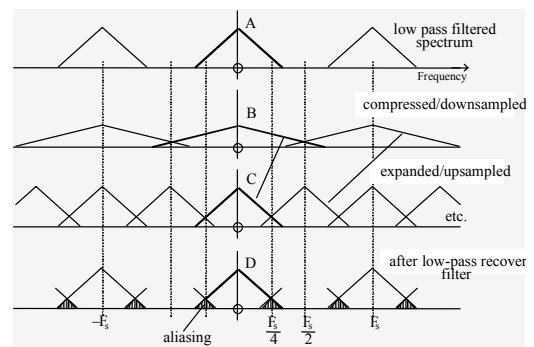


- $H_0(z)$: the z-transform of the low-pass analysis filter
- $H_1(z)$: the z-transform of high-pass analysis filter
- $G_0(z)$ and $G_1(z)$ are the corresponding synthesis filters
- The downsampling factor is 2, so as the upsampling

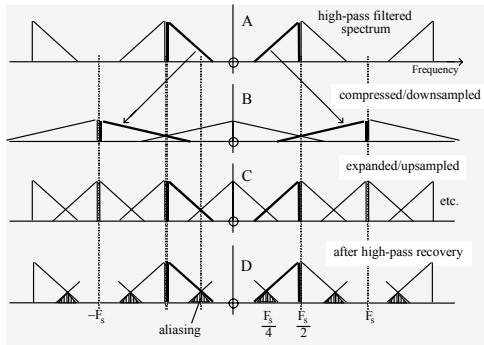
Subband Coding



Low-pass Subband Generation and Recovery



High-pass Subband Generation and Recovery



Example: Two-channel Filter Bank with Perfect Reconstruction

- Analysis filter impulse responses:
- Frequency responses:

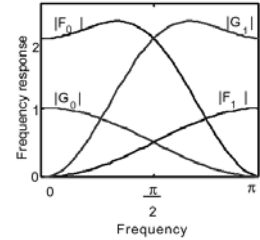
- Lowpass band:
 $\frac{1}{4}(-1, +2, +6, +2, -1)$

- Highpass band:
 $\frac{1}{4}(+1, -2, +1)$

- Synthesis filter impulse responses:

- Lowpass band:
 $\frac{1}{4}(+1, +2, +1)$

- Highpass band:
 $\frac{1}{4}(+1, +2, -6, +2, +1)$



Perfect Reconstruction in Subband Coding

The reconstructed output in z-transform notation:

$$Y(z) = G_0(z) \cdot Y_0(z) + G_1(z) \cdot Y_1(z)$$

where $Y_0(z) = \frac{1}{2}[H_0(z) \cdot X(z) + H_0(-z) \cdot X(-z)]$

$$Y_1(z) = \frac{1}{2}[H_1(z) \cdot X(z) + H_1(-z) \cdot X(-z)]$$

where the aliasing components from the downsampling of the lower and higher bands are given by $H_0(-z)X(-z)$ and $H_1(-z)X(-z)$ respectively

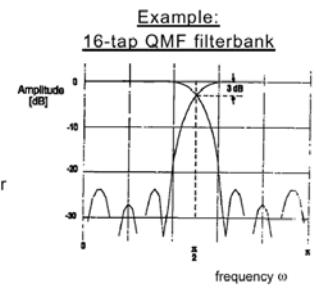
$$\begin{aligned} \text{The} \quad Y(z) &= \frac{1}{2}[H_0(z) \cdot G_0(z) + H_1(z) \cdot G_1(z)]X(z) \\ n \quad &+ \frac{1}{2}[H_0(-z) \cdot G_0(z) + H_1(-z) \cdot G_1(z)]X(-z) \end{aligned}$$

Quadrature Mirror Filter

- QMFs achieve aliasing cancellation by choosing

$$\begin{aligned} F_1(\omega) &= F_0(\omega + \pi) \\ &= -G_1(\omega) = G_0(\omega + \pi) \end{aligned}$$

- Highpass band is the mirror image of the lowpass band in the frequency domain



Perfect Reconstruction in Subband Coding

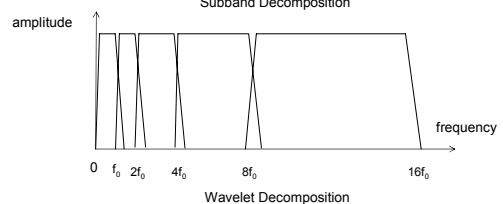
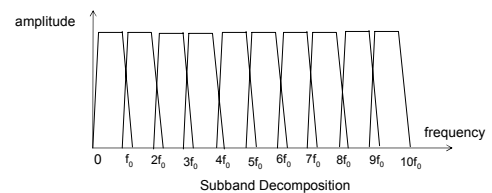
The first term is the desired reconstructed signal, while the second term is aliased components

So, sets $G_0(z) = H_1(-z)$ and $G_1(z) = -H_0(-z)$

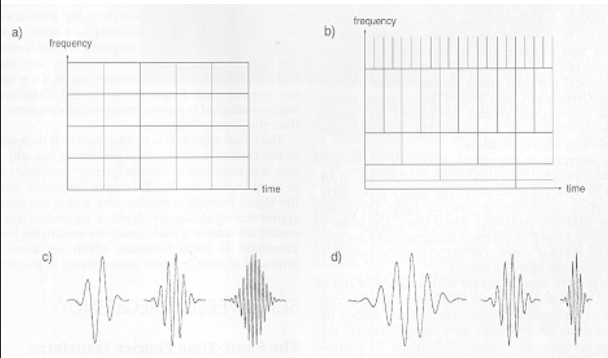
With such a relation between the synthesis and analysis filters, the reconstructed signal now becomes:

$$Y(z) = \frac{1}{2}[H_0(z) \cdot H_1(-z) - H_0(-z) \cdot H_1(z)]X(z)$$

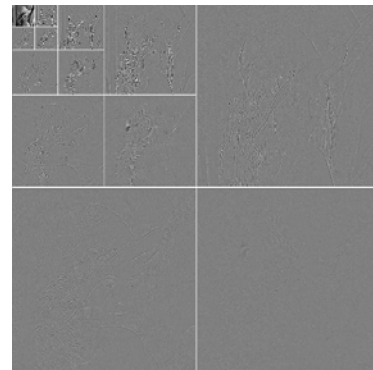
Subband and Wavelet Coding



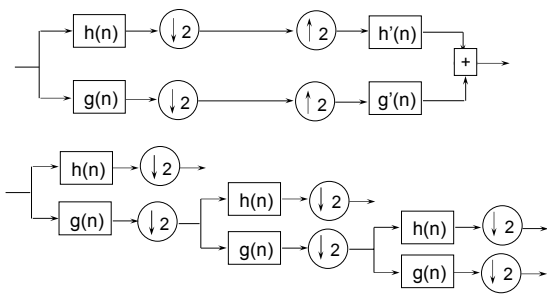
Subband and Wavelet Coding (Cont.)



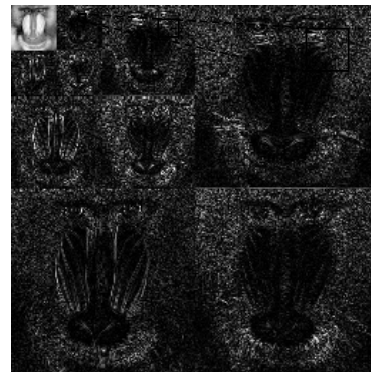
2-D Discrete Wavelet Transform (Cont.)



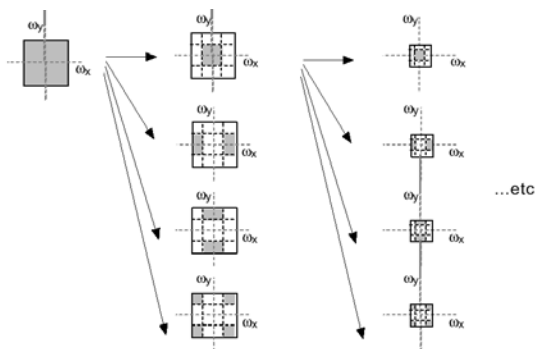
Subband and Wavelet Coding (Cont.)



Similarities among Image Subbands



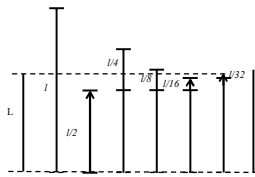
2-D Discrete Wavelet Transform



Zerotree Coding of Higher Bands

- The higher order wavelet coefficients are coded with the embedded zero-tree wavelet (EZW)
- The method
 - based on the concept of quantization by *successive approximation*,
 - exploits the similarities of the bands of the same orientation.

Principle of Successive Approximation



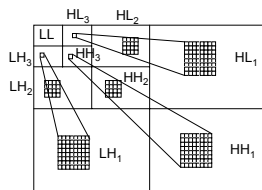
– As shown above, the quantized length, can be expressed as

Embedded Zerotree Wavelet Algorithm (Cont.)

- For the highest bands, ZTR and IZ symbols are merged into one symbol Z
- Successive approximation quantization and encoding
 - Initial "dominant" pass
 - Set initial threshold T, determine significant coefficients
 - Arithmetic coding of symbols ZTR, IZ, POS, NEG
 - Subordinate pass
 - Refine magnitude of coefficients **found significant so far** by one bit (subdivide magnitude bin by two)
 - Arithmetic coding of sequence of zeros and ones.
 - Repeat dominant pass
 - Set previously found significant coefficients to zero
 - Decrease threshold by factor of 2, determine new significant coefficients
 - Arithmetic coding of symbols ZTR, IZ, POS, NEG
 - Repeat subordinate and dominate passes, until bit budget is exhausted.

Zerotree Coding: Quadtree Representation of Higher Bands

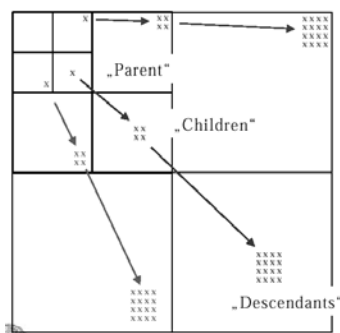
- Subimages of lower bands have quarter dimensions of their higher bands
- A quad-tree representation of the bands of the same orientation for a 10-band splitting is shown below (three-stage wavelet transform)
- If a coefficient in LH_3 is zero, it's more likely that its children in higher bands of LH_2 and LH_1 , will also be zero => "zero tree"



A Coding Example of EZW (Shapiro 93)

63	-34	49	10	7	13	-12	7
-31	23	14	-13	3	4	6	-1
15	14	3	-12	5	-7	3	9
-9	-7	-14	8	4	-2	3	2
-5	9	-1	47	4	6	-2	2
3	0	-3	2	3	-2	0	4
2	-3	6	-4	3	6	3	6
5	11	5	-6	0	3	-4	4

Embedded Zerotree Wavelet Algorithm



- Idea: Conditional coding of all descendants (incl. children)
- Coefficient magnitude > threshold: significant coefficients
- Four cases
 - ZTR: zero-tree, coefficient and all descendants are not significant
 - IZ: coefficient is not significant, but some descendants are significant
 - POS: positive significant
 - NEG: negative significant

A Coding Example of EZW (Cont.)

Subband	Coef Value	Symbol	Rec Value
LL3	63	POS	48
HL3	-34	NEG	-48
LH3	-31	IZ	0
HH3	23	ZTR	0
HL2	49	POS	-48
HL2	10	ZTR	0
HL2	14	ZTR	0
HL2	-13	ZTR	0
LH2	15	ZTR	0
LH2	14	IZ	0
LH2	-9	ZTR	0
LH2	-7	ZTR	0
HL1	7	Z	0
HL1	13	Z	0
HL1	3	Z	0
HL1	4	Z	0
LH1	-1	Z	0
LH1	47	POS	48
LH1	-3	Z	0
LH1	-2	Z	0

Features of The EZW Algorithm

- The use of zero trees reduces the number of symbols to be encoded
- The use of a very small alphabet to represent an image makes adaptive arithmetic coding very efficient
- The maximum distortion of each pass is bounded by the threshold used in the pass
- At any given pass, only the coefficients with magnitudes larger than the current threshold are encoded nonzero => prioritize the coefficients of different importance
- With the successive approximation process, the encoding and decoding can stop at any point => not only makes possible an extremely precise bit-rate control, but also achieves the best possible quality at a given bit-budget
- Embedded spatial/SNR scalability

Set Partitioning in Hierarchical Trees (SPIHT)

- A. Said, A., W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE T-CSVT*, vol. 63, pp. 243-250, June 1996.
- JPEG-2000 standard is similar to SPIHT (EBCOT)
- Preliminary Algorithm
 - Output $n = \text{floor}(\log_2(\max_{(i,j)}\{|C_{i,j}|\}))$
 - Output μ_n , followed by the pixel coordinate $\eta(k)$ and sign of each of the coefficients such that $2^n \leq |C_{\eta(k)}| \leq 2^{n+1}$ (**sorting pass**)
 - Output the n th MSB of all the coefficients with $|C_{i,j}| \geq 2^n$, in the same order used to send the coordinates (**refinement pass**)
 - Decrease n by one and go to step (2)

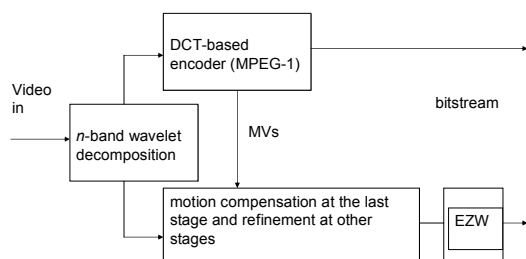
Embedded Zerotree Wavelet Algorithm (Cont.)

- Decoding: bitstream can be truncated to yield a coarser approximation: "embedded" representation
- Further details: J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445-3462, December 1993.
- Enhancement SPIHT coder: A. Said, A., W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 63, pp. 243-250, June 1996.
- JPEG-2000 standard similar to SPIHT

Set Partitioning in Hierarchical Trees (SPIHT)

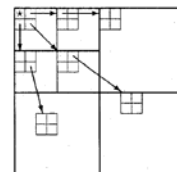
- Function used in the sorting pass
- Assume $O(i,j)$ represents a set of coordinates of all off-spring of node (i,j) , that is $O(i,j) = \{(2i,2j), (2i, 2j+1), (2i+1,2j), (2i+1,2j+1)\}$
- Define $D(i,j)$ as a set of coordinates of all descendants of the node (i,j) , and H , a set of coordinates of all spatial orientation tree
- Finally define $L(i,j) = D(i,j) - O(i,j)$

Video Coding with Wavelet Transform

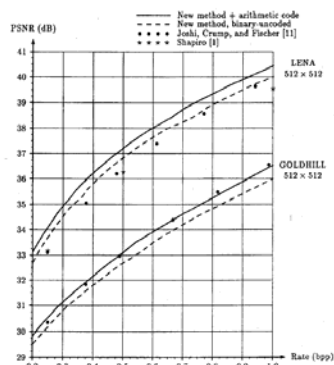


Set Partitioning Rules

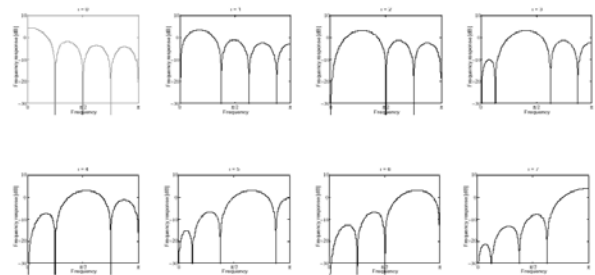
- the sets $\{(i,j)\}$ and $D(i,j)$, for all (i,j) "member of" H
- If $D(i,j)$ is significant, it is partitioned into $L(i,j)$ plus the four element sets with $(k,l) \in O(i,j)$
- If $L(i,j)$ is significant, it is partitioned into the four sets $D(k,l)$, with $(k,l) \in O(i,j)$
- Each of the four sets now has the format of the original set, and the sample partition can be used recursively



Performance of SPIHT



Frequency Response of a DCT of Order $N = 8$



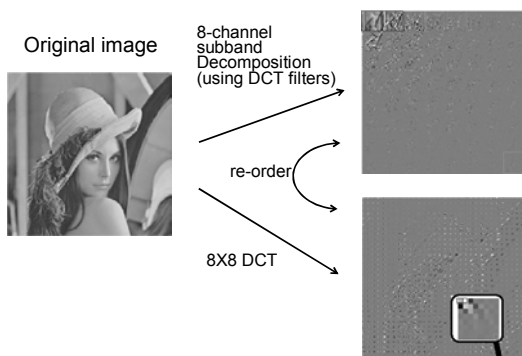
Subband Coding vs. Transform Coding

- Transform coding is a special case of subband coding with:
 - Number of bands = order of transform N
 - Subsampling factor $K = N$
 - Length of impulse responses of analysis/synthesis filters $\sim N$
- Filters used in subband coders are not in general orthogonal.

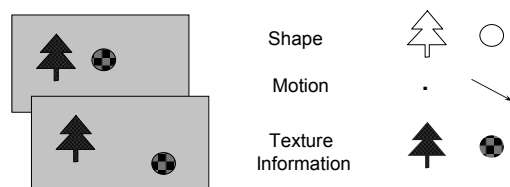
Model-based Coding

- Has potential to achieve very low bitrates by using source models at both the encoder and decoder
 - Object oriented coding
 - Wired frame model

Subband Coding vs. Transform Coding (Cont.)



Parameters of Object Based Coding



Wire-Frame Model

