ECE462 – Lecture 18

Embedded zerotree wavelet coder (EZW)

“Embedded Code”: One that contains all lower rate codes “embedded” at the beginning of the bitstream.

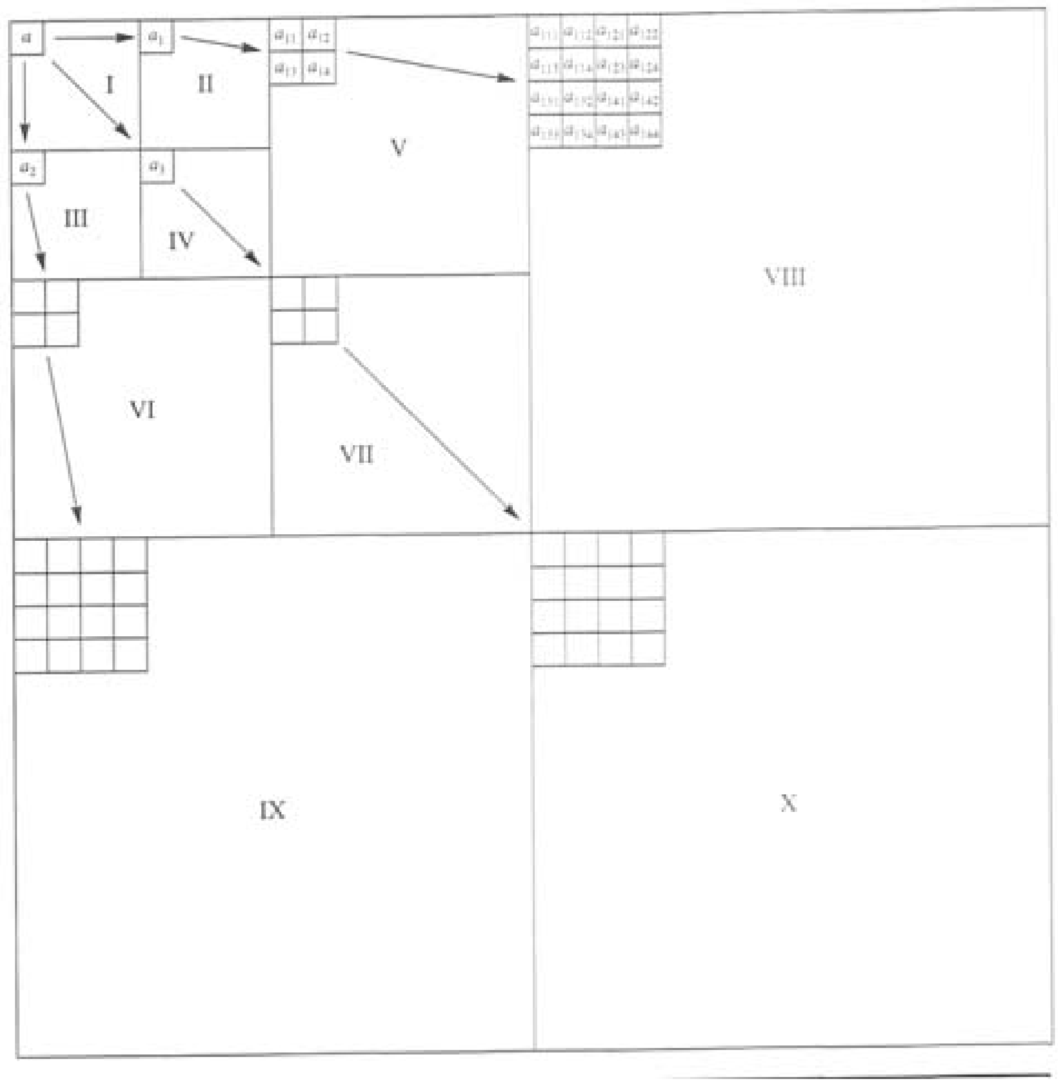
An embedded code allows the encoder to terminate the encoding at any point and thus meet the target bit rate exactly.

The decoder can stop to decode at any point and produce reconstructions corresponding to all lower rate encoding.

EZW Basic Idea

- There are wavelet coefficients on different sub-bands that represent the same spatial locations in the image. Then a single coefficient in a smaller sub-band may represent the same spatial location as multiple coefficients in other sub-bands.

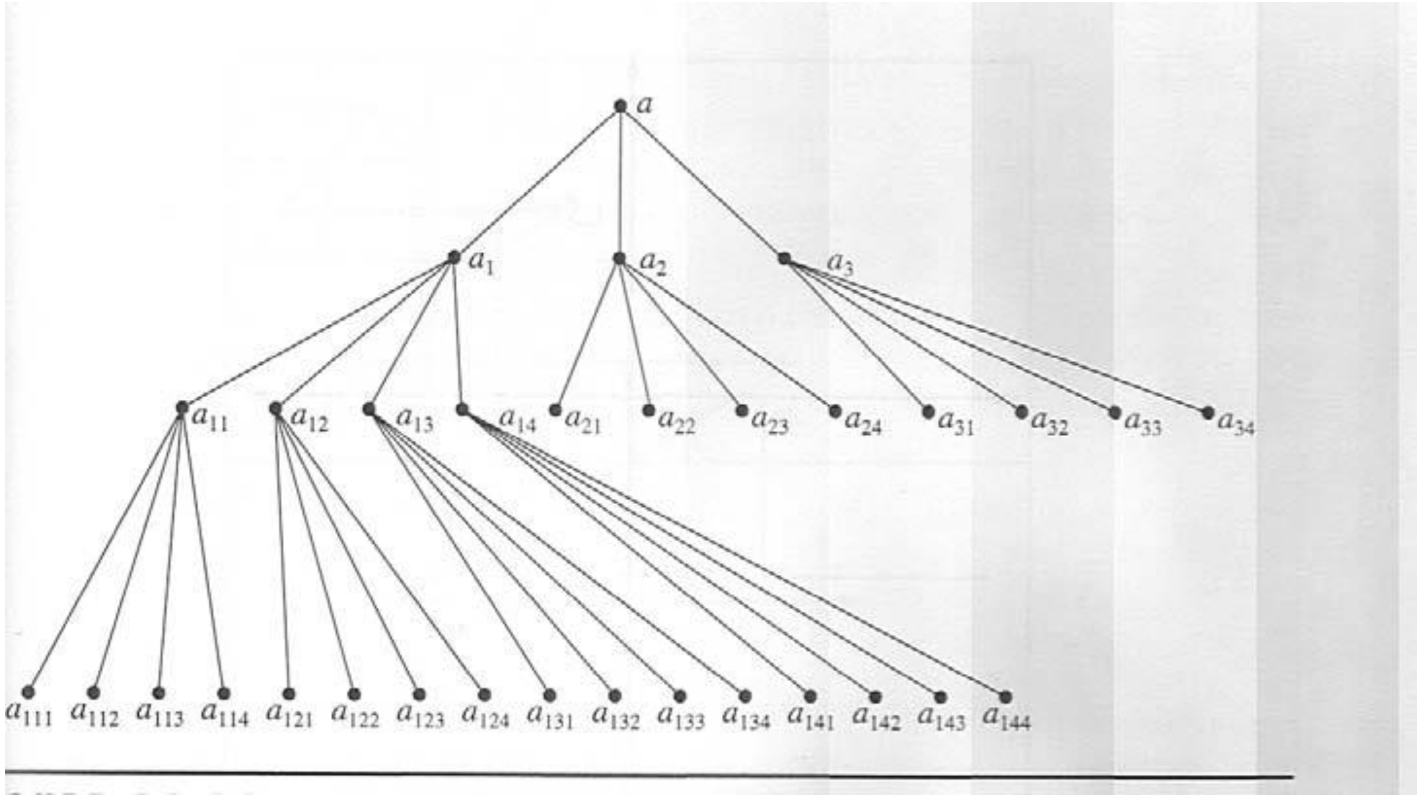
- 10-band wavelet decomposition (3-level)
  - The coefficient a forms the root of the tree with three descendants a_1,a_2,a_3
  - The coefficient a_1 has descendants a_{11}, a_{12}, a_{13}, a_{14} and so on
  - The coefficient a in band I represents the same spatial location as coefficients a_1 in band II, a_2 in band III, a_3 in band IV…



EZW Basic Idea

Total of 64 coefficients in this tree.

Data structure used in EZW ->

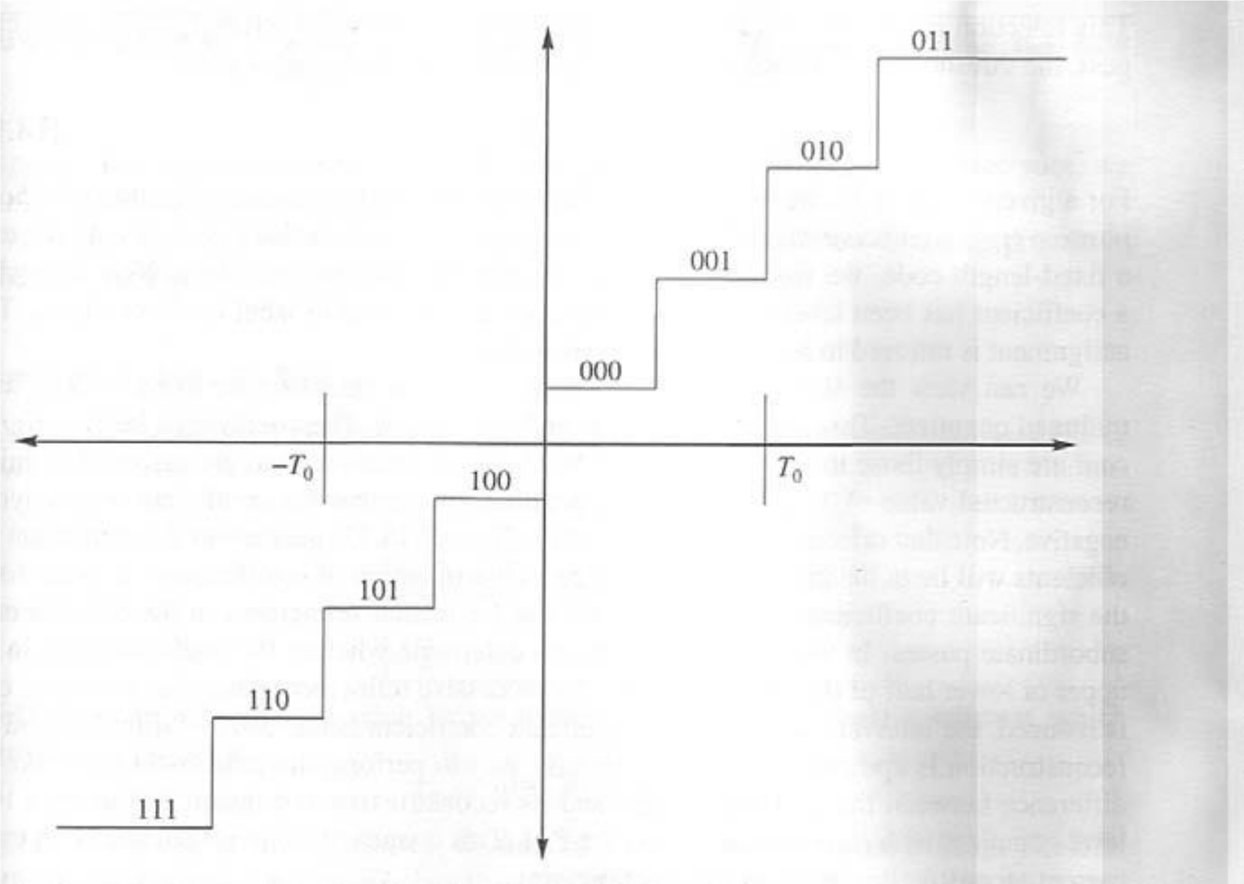


EZW Basic Idea

- In Wavelet decomposition of natural images most of the energy is compacted into lower bands. Thus, usually coefficients closer to the root of the tree have higher magnitudes than coefficients further away from the root.

- If we determine that all coefficients arising from a particular root are less than T_0 we can inform the decoder and use 2 bits/coefficient instead of 3 bits.

EZW Basic Idea



EZW Basic Idea

- The EZW algorithm accomplishes this by composing coefficients to a number of thresholds.

$$T_0, T_1 = \frac{T_0}{2}, T_2 = \frac{T_1}{2}, \dots.., T_i = \frac{1}{2} T_{i-1}$$

$$T_0 = 2^{\lfloor \log_2 C_{MAX} \rfloor}$$

Where C_{MAX} is the largest coefficient, and $\lfloor \quad \rfloor$ means integer part.

EZW Terminology

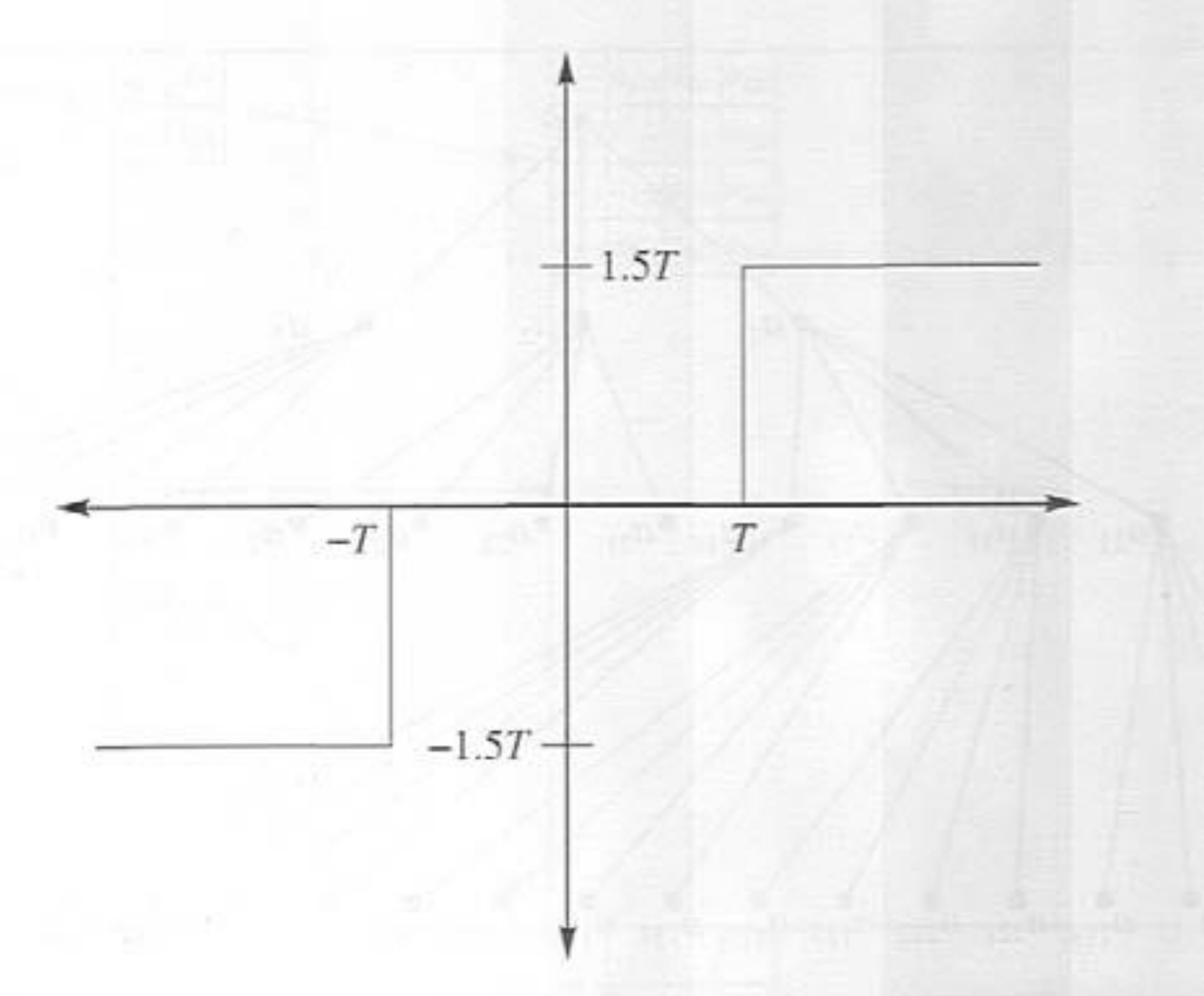
- Given a threshold T, a coefficient with magnitude greater than T is called “significant” at level T.
- If a coefficient at level T is insignificant and all its descendants have also magnitude less than T then it is called “zero tree root”.
- If a coefficient at level T is insignificant but some of its descendants have magnitude greater than T, it is called “isolated zero”.
- Each pass of the EZW consists of two steps:
  - Significant map encoding or dominant pass
  - Refinement or subordinate pass

EZW Terminology

- For a given value of T_i we assign one of four possible labels to coefficients
  - Significant positive (P)
  - Significant negative (n)
  - Zero tree root (t)
  - Isolated Zero (z)
- With a fixed length code we need 2 bits for each label
- If a coefficient is (t) we do not label its descendants.

EZW Terminology

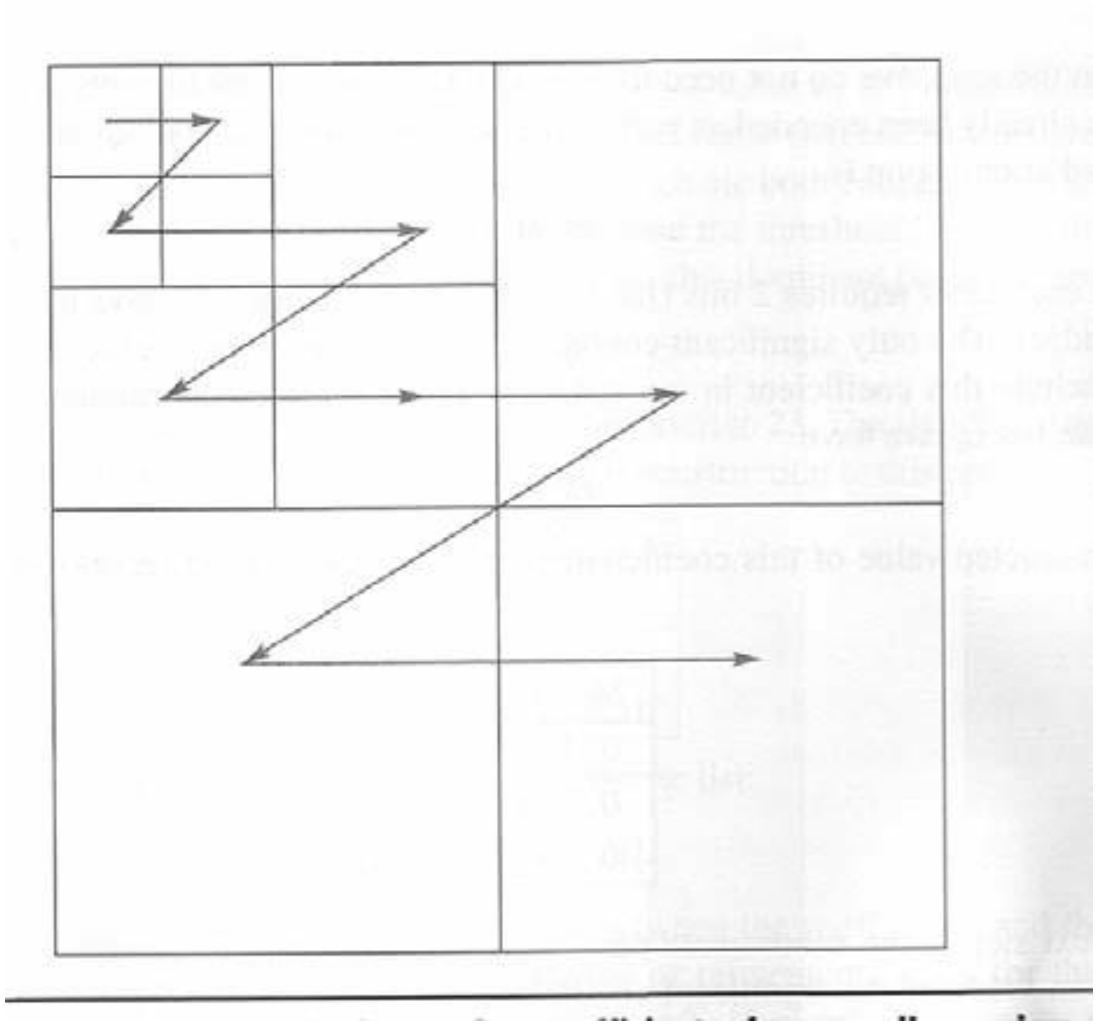
- Dominant pass coding can be viewed as 3-level quantization. “Significant” coefficients are assigned an initial reconstructed value of $1.5T_i$ or $-1.5T_i$. Then the coefficients are included into the “refinement list” for refinement during subordinate passes.



EZW Terminology

- In the refinement pass we determine if the coefficient is in $\left[T, \frac{3T}{2}\right]$ or $\left[\frac{3T}{2}, 2T\right]$ in order to be refined.
- In practice refinement is done as follows:

Take the difference between coefficient value and its reconstruction and quantize it to $\pm\frac{T}{4}$.



EZW Terminology

- Then, add the quantized value of current reconstruction.
- Coefficients that are not significant are scanned as with parent nodes scanned first.

Example to demonstrate EZW

- Consider the following wavelet decomposition

26	6	13	10
-7	7	6	4
4	-4	4	-3
2	-2	-2	0

Then, $T_0 = 2^{\lfloor \log_2 26 \rfloor} = 16$

Example to demonstrate EZW

- 1^{st} pass

Dominant: 26 > 16 so 26 is (p)

6 and its descendants (13,10,6,4) < 16 so 6 is (t)

Similarly -7 is (t)

and 7 is (t)

Thus, the dominant pass provides

D_0 : p t t t (8 bits)

The only significant coefficient is 26 so the refinement list is L_s={26}

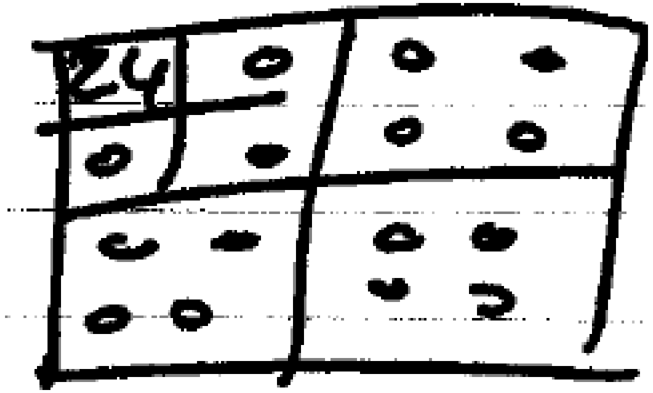
Example to demonstrate EZW

- The reconstructed value is 1.5 * T_0=24 and

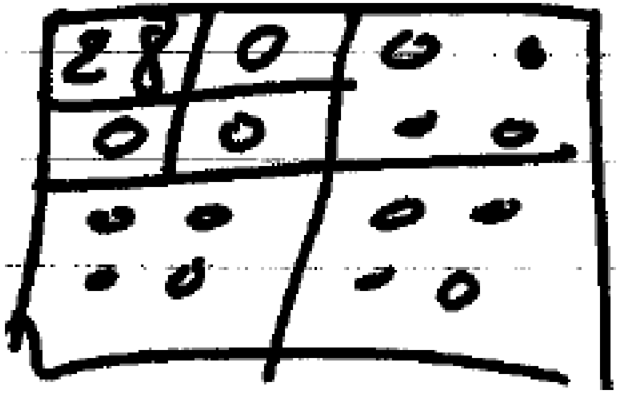
Subordinate Pass: 26-24 = 2 > 0

Correction term $\frac{T_0}{4} = 4$

(1 more bit to transmit correction)



To summarize, the 1st pass produces a reconstruction shown on the right and requires 9 bits in total.



Example to demonstrate EZW

*	6	13	10
-7	7	6	4
4	-4	4	-3
2	-2	-2	0

2nd Pass: Reduce threshold by 2: $T_1 = T_0/2 = 8$

Dominant Pass: Scan coefficients that have not been classified as significant as of yet.

6 < 8 but has descendants that are significant at this so 6 -> (z)

|-7| < 8 with no significant descendants so -7 -> (t)

Also, 7 -> (t)

Next, 13 -> (p), 10 -> (p), 6 -> (t), 4 -> (t)

Thus, D_1: z t t p p t t

Reconstruction of the new significant values is

1.5 . T_1 = 12 and L_s={26, 13, 10}

(14 bits)

28	0	12	12
0	0	0	0
0	0	0	0
0	0	0	0

Example to demonstrate EZW

26	0	14	10
0	0	0	0
0	0	0	0
0	0	0	0

Subordinate Pass: Correction vaue = $\pm\frac{T_1}{4} = \pm 2$

So 26-28 = -2 -> Correction = -2 (1 bit)

13-12 = 1 -> Correction = 2 (1 bit)

10-12 = -2 -> Correction = -2 (1 bit)

Thus, the second pass generates a reconstruction of and a total of

9 + 14 + 3 = 26 bits.

Example to demonstrate EZW

3rd Pass: Reduce threshold by 2, $T_2=T_1/2 = 4$

Dominant pass: Scan coefficients not classified significant as of yet.

Clearly, 6 -> (p)

-7 -> (n)

7 -> (p)

6 -> (p), 4 -> (p),

4 -> (p), -4 -> (n),

2 -> (t), -2 -> (t), 4 -> (p), -3 -> (t), -2 -> (t), 0 -> (t)

*	6	*	*
-7	7	6	4
4	-4	4	-3
2	-2	-2	0

Example to demonstrate EZW

Thus: D_2 : p n p p p p n t t p t t t

Reconstruction of the new significant values is 1.5 T_2=6

And L_s={26,13,10,6,-7,7,6,4,4,-4,4}

Subordinate pass: Correction of $\pm\frac{T_2}{4} = \pm 1$

26-26 = 0 correction of 1

13-14 = -1 correction of -1

10-10 = 0 correction of 1

6-6 = 0 correction of 1

-7 – (-6) = -1 correction of -1

C26 bits

26	6	14	10
6	6	6	6
6	-6	6	0
0	0	0	0

Example to demonstrate EZW

7-6 = 1 correction of 1

6-6 = 0 correction of 1

4-6 = -2 correction of -1

4-6 = -2 correction of -1

-4 – (-6) = 2 correction of 1

4-6 = -2 correction of -1   (11 bits)

So the third pass generates a reconstruction of and a total of

26 + 26 + 11 = 63 bits.