# Multiresolution Analysis and Image Compression

Statistical Signal Processing

Arijit Das

# Outline

- Multiresolution Analysis
  - Orthogonal Wavelets
  - Bi-orthogonal Wavelets
  - Decomposition and Reconstruction
- Set Partition In Hierarchical Trees
  - Algorithm
  - Example

# Multiresolution Signal Decomposition

- Analyze the signal at different frequencies with different resolutions
- Unlike Fourier Analysis, multiresolution analysis uses basis formed by functions that are finite in both frequency as well as spatial domain
- Good time resolution and poor frequency resolution at high frequencies
- Good frequency resolution and poor time resolution at low frequencies
- More suitable for short duration of higher frequency and longer duration of lower frequency components

# Definition and Properties

A multiresolution function is sequence of subspaces $\{V_j\}$ of $L^2(R)$ satisfying

$$\mathbf{V_j} \subset \mathbf{V_{j+1}} \quad \forall \mathbf{j} \in \square\ , \mathbf{V_{-\infty}} = \mathbf{\{0\}}, \quad \mathbf{V_{\infty}} = \mathbf{L^2}$$

$$\text{clos}_{L^2(\mathbb{R})}\left(\bigcup_{j\in\mathbb{Z}} V_j\right) = L^2(\mathbb{R}),$$

$$\bigcap_{j\in\mathbb{Z}} V_j = \{0\};$$

$$f \in V_j \Longleftrightarrow f(2\cdot) \in V_{j+1}, \qquad j \in \mathbb{Z};$$

$$f \in V_0 \Longrightarrow f(\cdot - k) \in V_0, \qquad k \in \mathbb{Z}.$$

- If $\varphi(t)$ is in $V_0$ , it is also in $V_1$ ( a space spanned by ) $\varphi(2t)$
- Hence $\varphi(t)$ can be expressed as weighted sum of shifted $\varphi(2t)$

$$\varphi(t) = \sum h(n) 2^{\frac{1}{2}} \varphi(2t - n), \qquad n \in \mathbb{Z}$$

where coefficients h(n) are a sequence of real (or complex) numbers called the scaling function coefficients ( or scaling filter ). This recursive equation is fundamental to the theory of scaling functions.

- The scaling function is important because it shows that any function of $V_{n+1}$ is also defined in $V_n$ while the reverse is not true.
- Focusing on the nth subspace level of the nested subspaces, $V_{n+1}$ subset of $V_n$ implies that there is a subspace $W_{n+1}$ which is the complement of $V_{n+1}$ so that $V_n$ can be decomposed as

$$V_n = W_{n+1} \oplus V_{n+1}$$

- As a result the multiresolution function can be represented as two coefficient vectors one defined in $W_{n+1}$ and the other in $V_{n+1}$

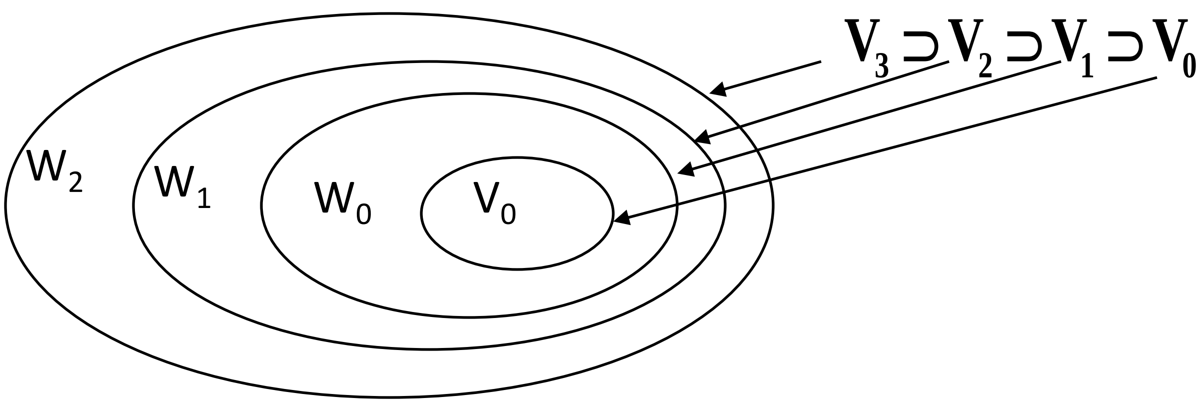
$$\mathbf{V_0 \subset V_1 \subset V_2 \subset \ldots \subset L^2}$$

$$\mathbf{V_1 = V_0 \oplus W_0}$$

$$\mathbf{V_2 = V_1 \oplus W_1 = V_0 \oplus W_0 \oplus W_1}$$

$$\mathbf{L^2 = V_0 \oplus W_0 \oplus W_1 \oplus \ldots}$$

$$\mathbf{W_2 \perp W_1 \perp W_0 \perp V_0}$$



- Scaling function : $\phi(x)$

  Define : $\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k)$

  Therefore $V_j = \underset{k}{span}\left\{\phi_{j,k}(x)\right\}$

  $V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_\infty$

  since $\phi(x) = \sum h_\phi(n)\sqrt{2}\phi(2x \quad n)$

- Wavelet function : $\psi(x)$ span *difference* between adjacent scale

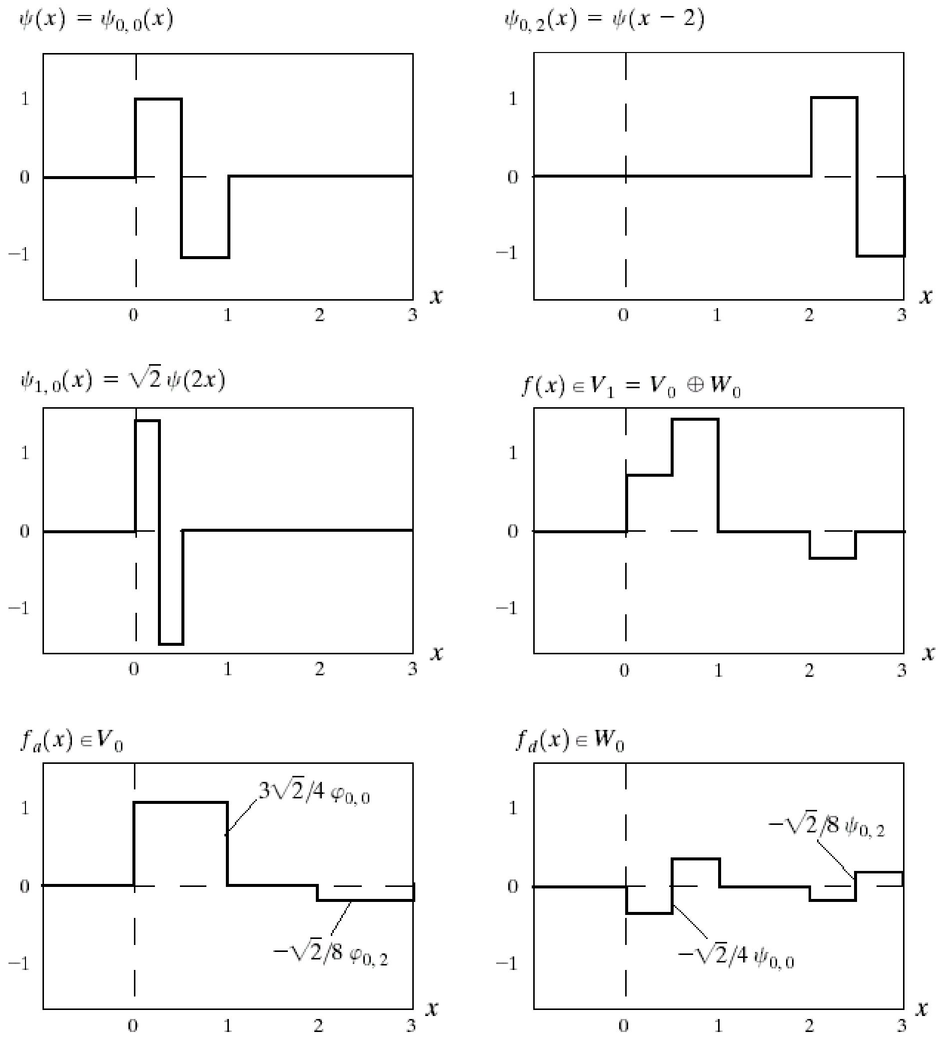
  Define : $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$

  Therefore $W_j = \underset{k}{span}\left\{\psi_{j,k}(x)\right\}$

  $W_j \subset V_{j+1}$

  since $\psi(x) = \sum_n h_\psi(n)\sqrt{2}\phi(2x - n)$

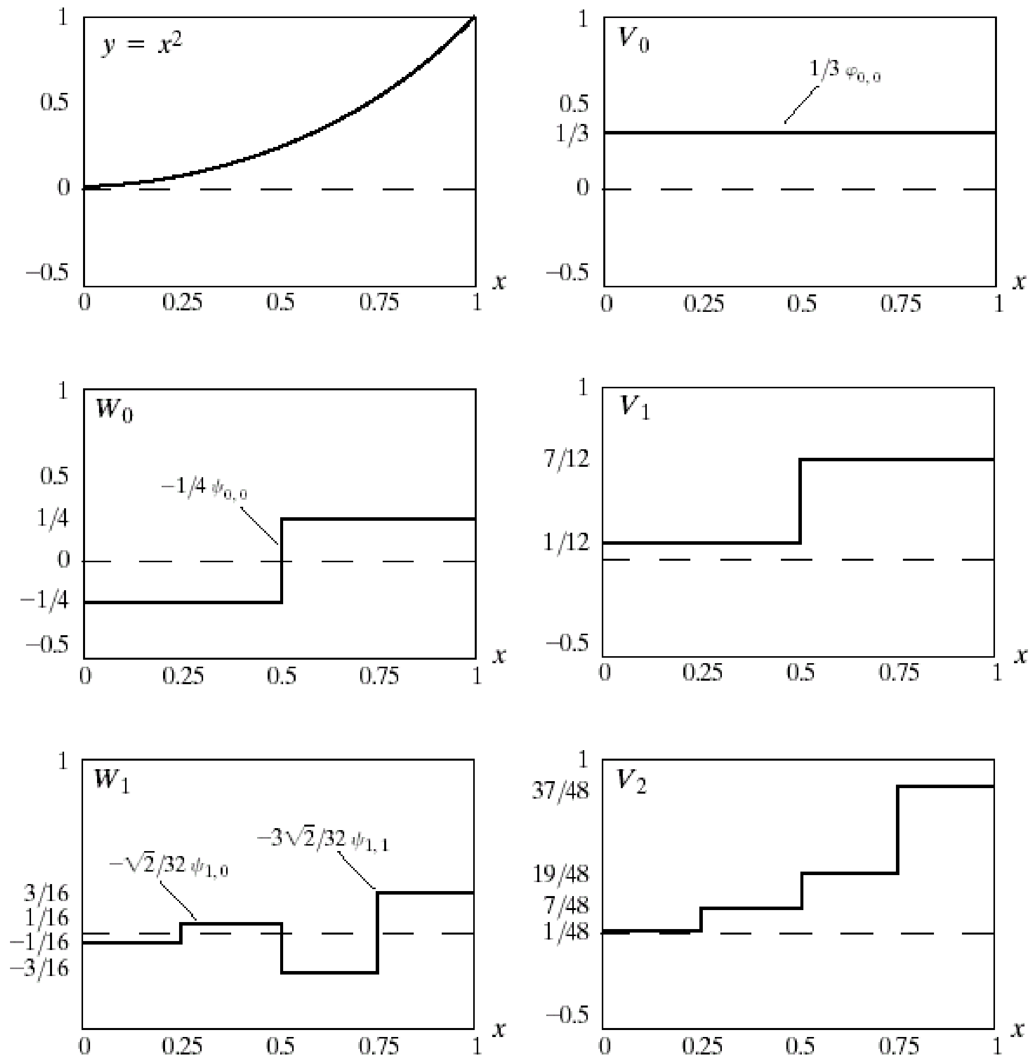
# Wavelet Functions




a b
c d
e f

**FIGURE 7.12** Haar wavelet functions in $W_0$ and $W_1$.

# Wavelet Function



# Orthogonal wavelet bases

- Find an orthogonal basis of $V_j$ : $\varphi_{j,k}(t) = 2^{j/2}\varphi(2^j t - k),\ k \in \mathbb{Z}$
- Two-scale equations :

$$\varphi_{j,k} = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} h_n \varphi_{j+1,2k+n} \qquad \psi(t) = \sum_{n \in \mathbb{Z}} g_n \varphi(2t - n)$$

- orthogonality requires : $g_n = (-1)^n h_{1-n}$ $\qquad \sum_n h_n = 2$

$$\sum_n h_n h_{n+2k} = 2 \text{ if } k = 0\text{, otherwise} = 0$$

$$\sum_n (-1)^n n^m h_n = 0,\ m = 0, \ldots, N-1$$

N : number of vanishing moments of the wavelet function

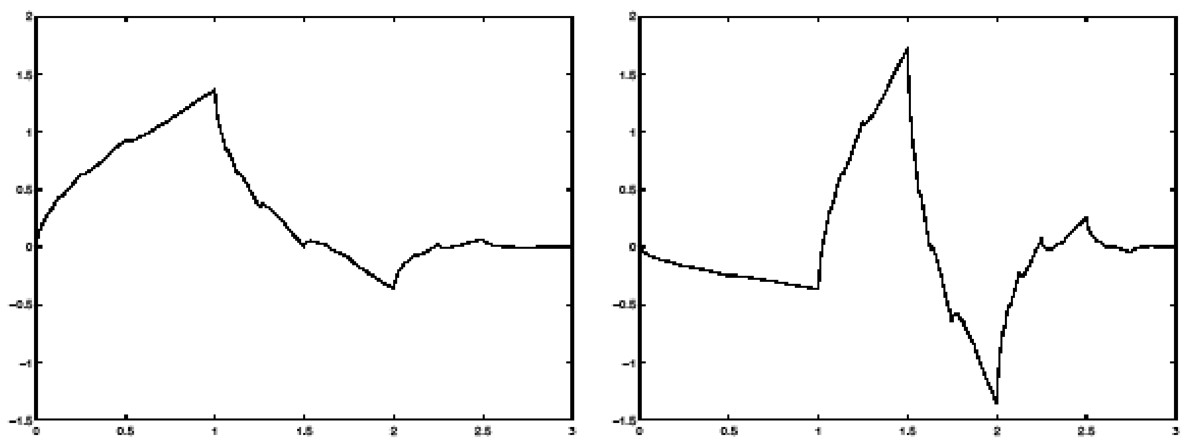
# Orthogonal wavelet bases

- Other way around , find a set of coefficients $h_n$ that satisfy the above equations.

Since the solution is not unique, other favorable properties can be asked for : compact support, regularity, number of vanishing moments of the wavelet function.

- then solve the two-scale equations.
- Example : Daubechies seeks wavelets with minimum size compact support for any specified number of vanishing moments.

The Daubechies D2 scaling and wavelet functions



$$h_n = ( \; \frac{1}{4}(1+\sqrt{3}) \; \frac{1}{4}(3+\sqrt{3}) \; \frac{1}{4}(3-\sqrt{3}) \; \frac{1}{4}(1-\sqrt{3})$$

# Bi-orthogonal Wavelet Transform :

It is a generalization of the orthogonal wavelets. Two other spaces $\tilde{O}_j$ and $\tilde{V}_j$ are introduced for the reconstruction:

- $V_{j-1} = V_j \oplus O_j$, and $V_j \not\perp O_j$
- $\tilde{V}_{j-1} = \tilde{V}_j \oplus \tilde{O}_j$, and $\tilde{V}_j \not\perp \tilde{O}_j$
- $\tilde{O}_j \perp V_j$ and $O_j \perp \tilde{V}_j$

# Bi-orthogonal Wavelet Transform :

Using two other filters $\tilde{h}$ and $\tilde{g}$, defined to be conjugate to $h$ and $g$. The reconstruction of the signal is performed with:

$$c_{j,k} = 2 \sum_{l} [c_{j+1,l} \tilde{h}(k - 2l) + w_{j+1,l} \tilde{g}(k - 2l)]$$

In order to get an exact reconstruction, two conditions are required for the conjugate filters:

- *Dealiasing condition*: $\hat{h}(\nu + \frac{1}{2})\hat{\tilde{h}}(\nu) + \hat{g}(\nu + \frac{1}{2})\hat{\tilde{g}}(\nu) = 0$
- *Exact restoration:* $\hat{h}(\nu)\hat{\tilde{h}}(\nu) + \hat{g}(\nu)\hat{\tilde{g}}(\nu) = 1$

→ The structure of the filter bank algorithm is the same.

# Fast algorithms

- we start with $f = \sum_{k=0}^{2^J - 1} c_{J,k}\varphi_{J,k}$

- we want to obtain $f = c_{0,0}\varphi + \sum_{j=0}^{J-1} \sum_{k \in \mathbb{Z}} d_{j,k}\psi_{j,k}$

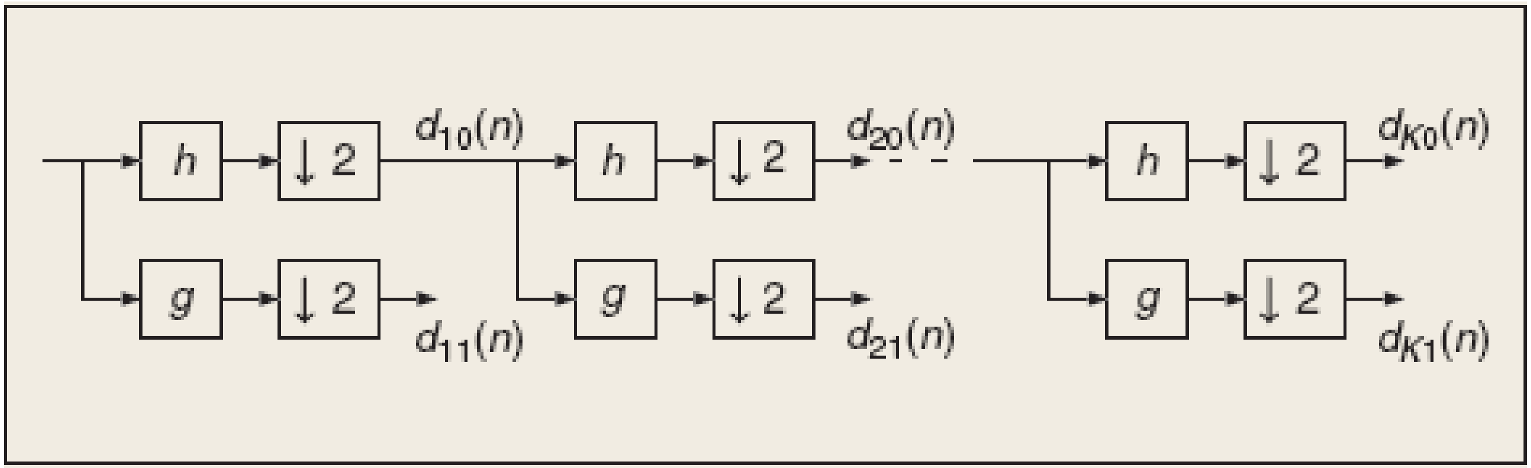
- we use the following relations between coefficients at different scales:

$$c_{j,k} = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} n_{n-2k} c_{j+1,n} \qquad d_{j,k} = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} g_{n-2k} c_{j+1,n}$$

- reconstruction is obtained with :

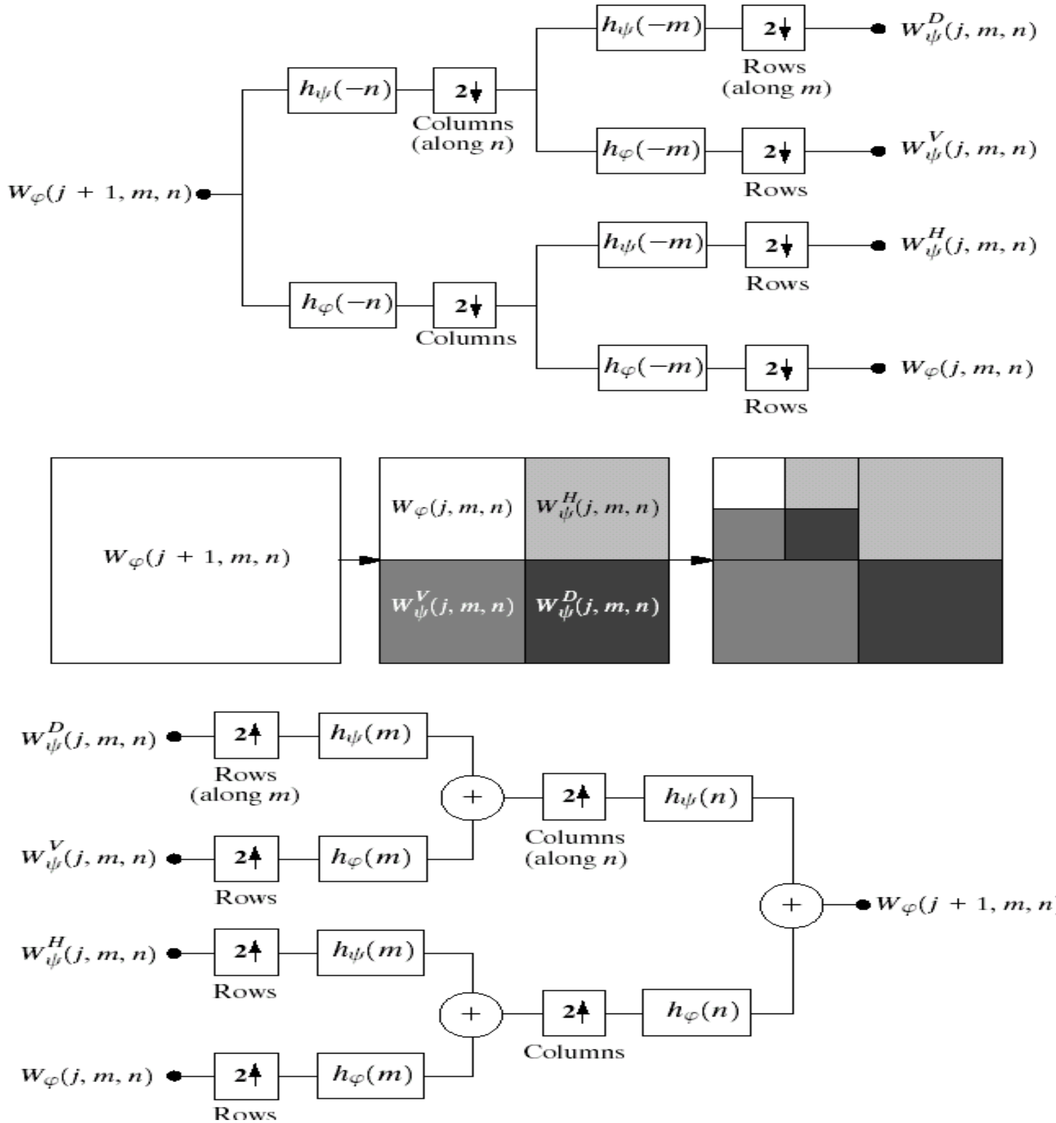
$$c_{j+1,k} = \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} c_{j,n} h_{k-2n} + \frac{1}{\sqrt{2}} \sum_{n \in \mathbb{Z}} d_{j,n} g_{k-2n}$$

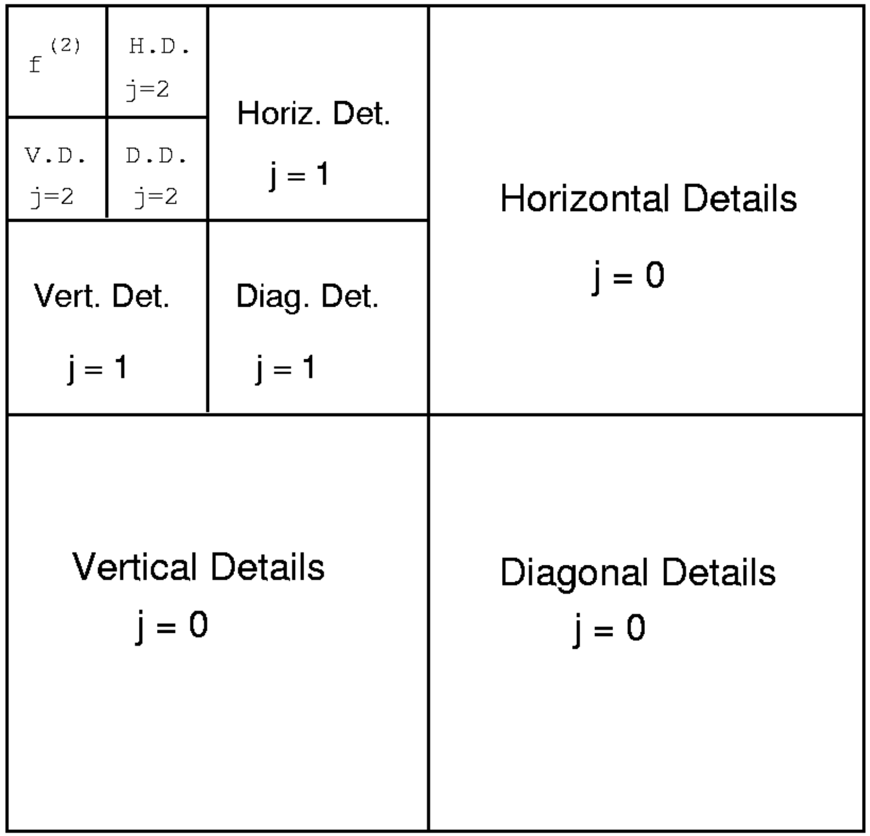
# Structure of the Wavelet Transform




▲ *1. A K-level, 1-D wavelet decomposition. The coefficient notation $d_{ij}(n)$ refers to the jth frequency band (0 for low and 1 for high) of the ith level of the decomposition.*

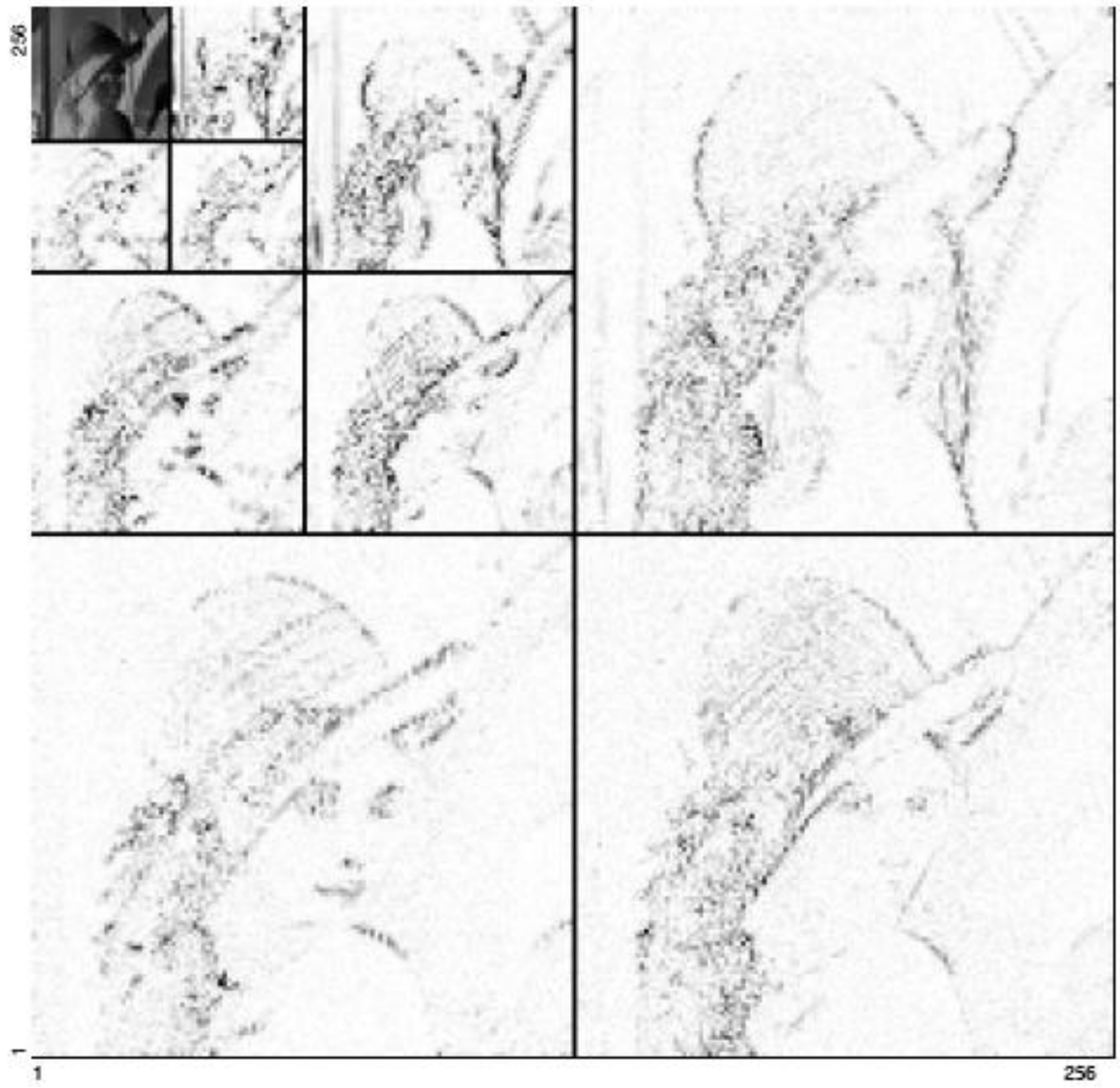
# 2-D Wavelet Transform




f (2)
H.D.
j=2
V.D.
j=2
D.D.
j=2
Horiz. Det.
j = 1
Vert. Det.
j = 1
Diag. Det.
j = 1
Horizontal Details
j = 0
Vertical Details
j = 0
Diagonal Details
j = 0

# Example :

# Example :

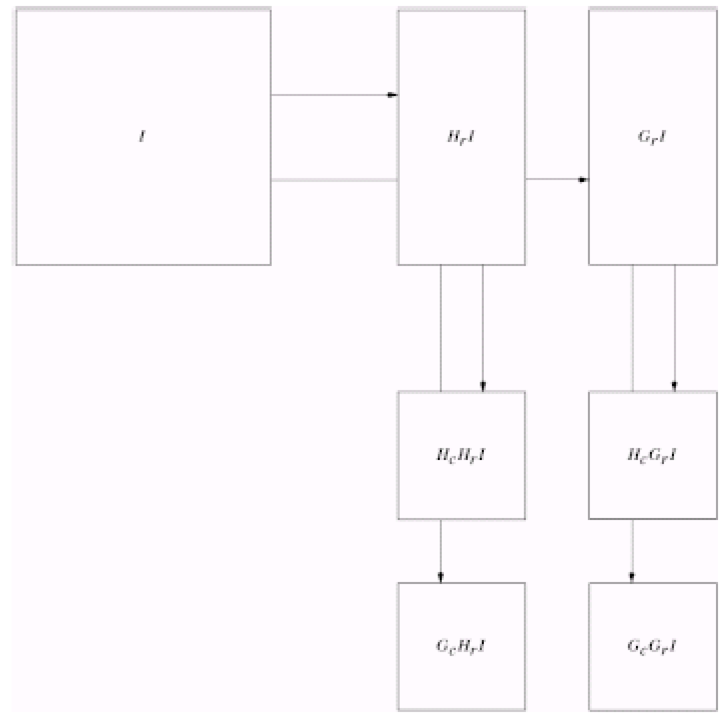


# SET PARTITION IN HIERARCHICAL TREES (SPIHT)

# SPIHT

- Uses spatially oriented tree to describe the relationship between
  - The parents on higher levels
  - The children and the grandchildren on the lower levels

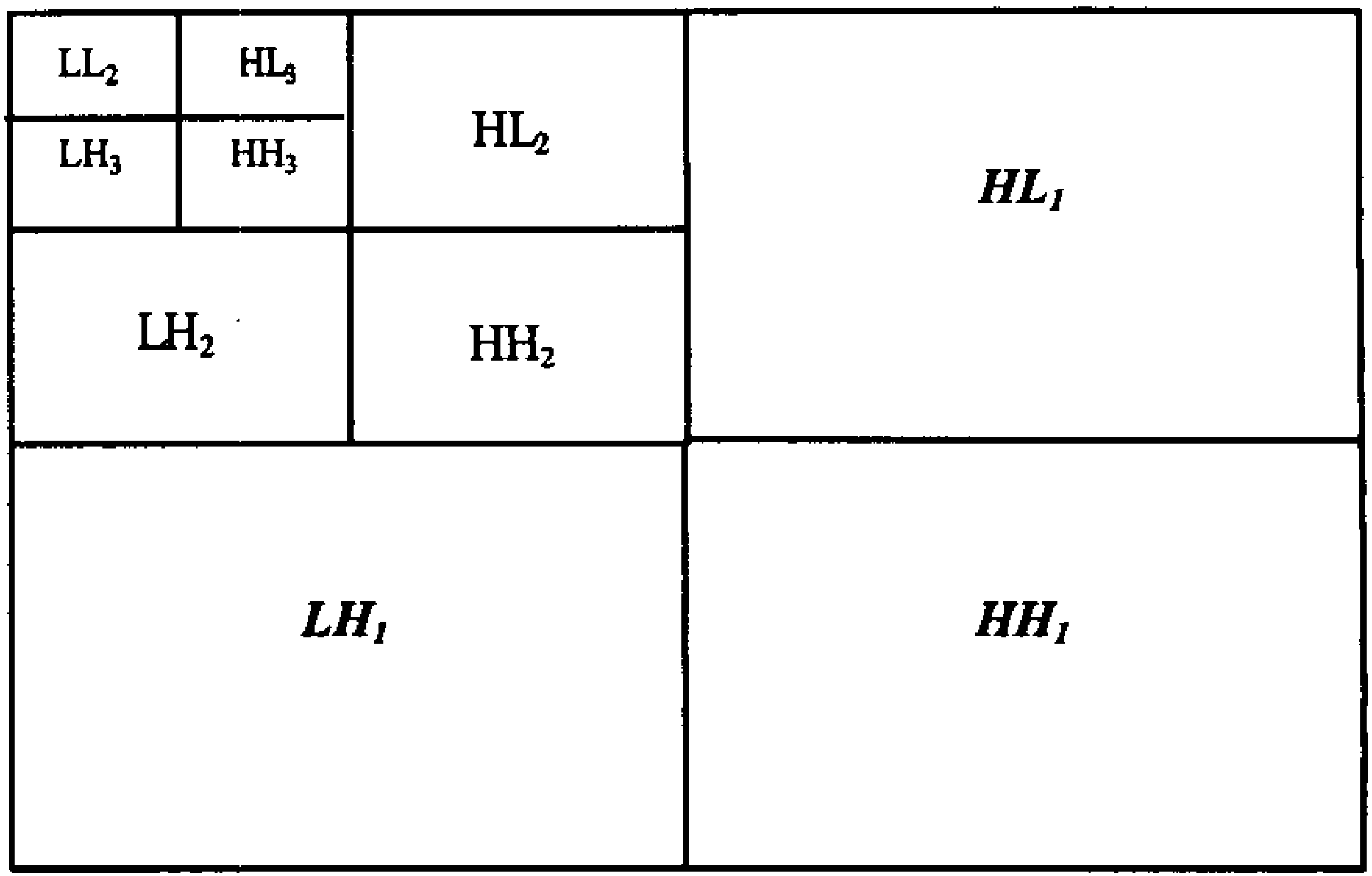
# The Wavelet Transform



# The Wavelet Transform

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 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 |

# Wavelet



# SPIHT Algorithm

Let: *O(i,j)*: Set of coordinates of all the offspring of node (i,j).

*D(i,j)*: Set of coordinates of all descendants of node (i,j).

*H*: Set of coordinates of all spatial orientation tree roots (nodes in the highest pyramid level).

$L(i,j) = D(i,j) - O(i,j)$

*Type A is pixels* $\in D(i,j)$

*Type B pixels* $\in L(i,j)$.

*LIS is a list of the insignificant set*

*LIP is a list of the insignificant pixels*

*LSP is a list of the significant pixels*

$$S_n(T) = \begin{cases} 1, & \max\{|c_{i,j}|\} \geq 2^n \\ 0, & otherwise \end{cases} \quad (7)$$

The steps of the algorithm are as follows:

1) **Initialization:** The number of bits needed to represent the magnitude of the coefficients $c_{i,j}$ is given by:

$$n = \lfloor \log_2 (\max_{(i,j)} \{ | c_{i,j} | \} \rfloor \quad (8)$$

Set the LSP as an empty set, and all (i,j) $\in H$ to LIS as Type A entries.

2) **Sorting Pass:**

**2.1** For each entry (i,j) in the LIP do the following:

**2.1.1** Transmit $S_n(i,j)$

**2.1.2** If $S_n(i,j)$ =1;them move (i,j) to LSP and transmit the sign of $c_{i,j}$.

**2.2** For each entry (i,j) in the LIS do:

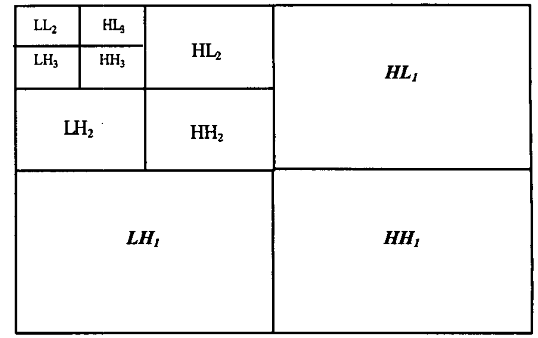
**2.2.1** If the entry is of type A then

- Transmit $S_n(D(i,j))$.
- If $S_n(D(i,j))$=1 then

**i.** each $(k,l) \in O(i,j)$ do:

- Output $S_n(k,l)$;
- If $S_n(k,l)$=1 then add (k,l) to the LSP and output the sign of $c_{k,l}$.
- If $S_n(k,l)$=0 then add (k,l) to the end of the LIP.

ii. If $L(i,j) \neq 0$, then move (i,j) to the end of the LIS as an entry of type B, and go to step 2.2.2; otherwise, remove entry (i,j) from the LIS.




**2.2.2** If the entry is of type B then

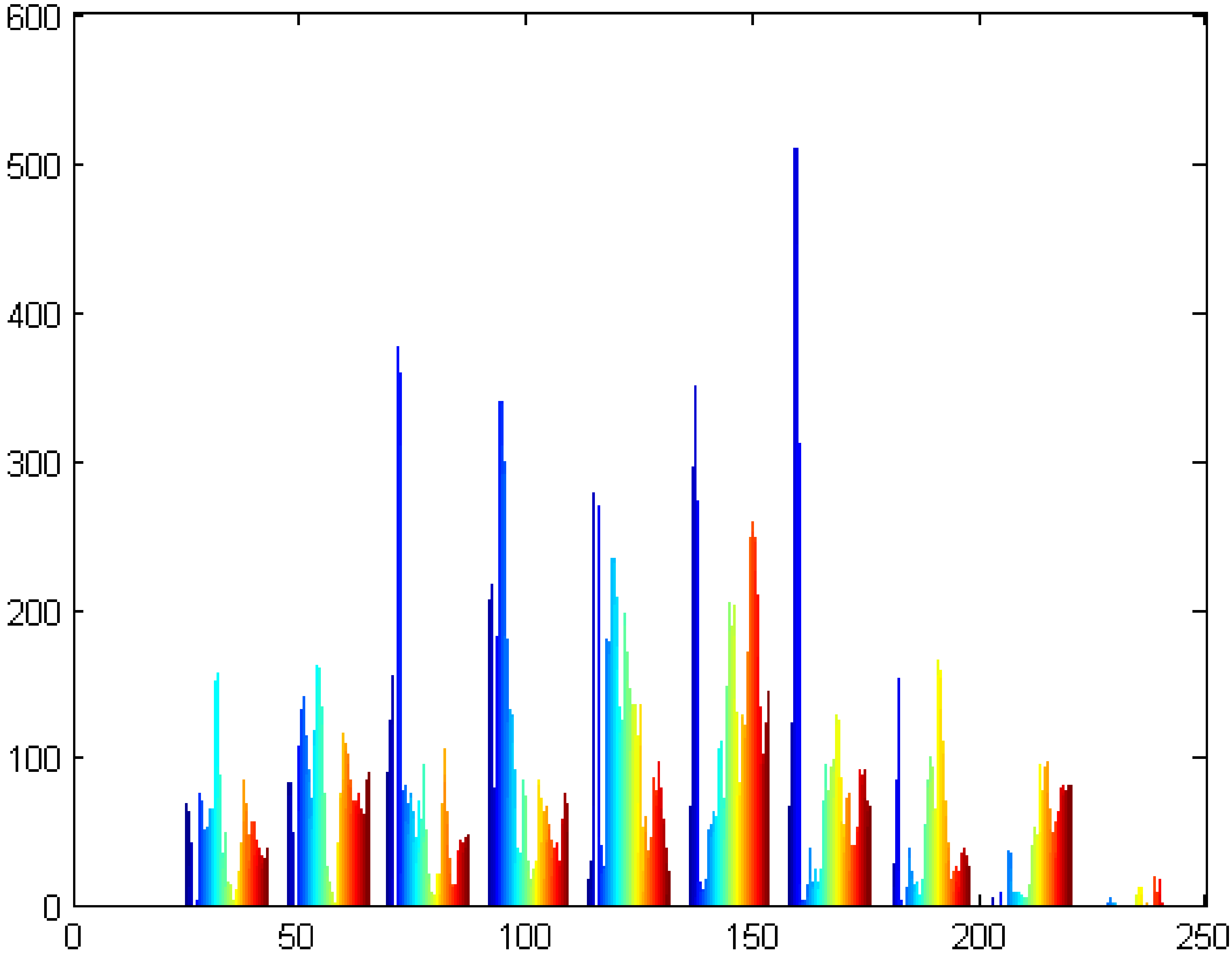
- Transmit $S_n(L(i,j))$.
- If $S_n(L(i,j))=1$ then

  i. Add each$(k,l) \in O(i,j)$ to the end of the LIS as an entry of type A.

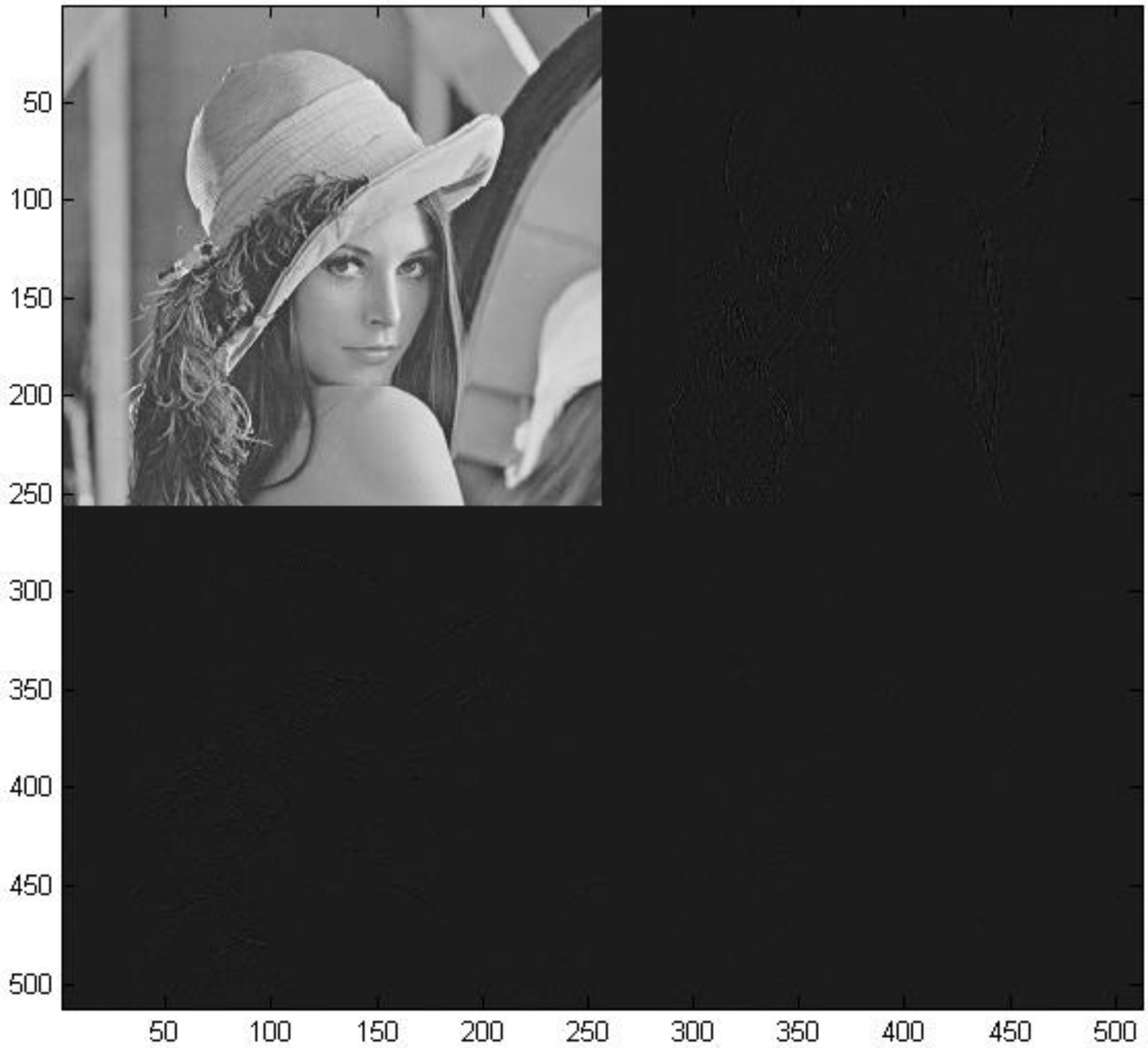
  ii. Remove $(i,j)$ from the LIS.

3) **Refinement Pass:** for each entry $(i,j)$ in the LSP, except those included in the last sorting pass.(i.e. with the same n), output the nth most significant bit of $|c_{i,j}|$.
4) **Quantization step update:** Decrement n by 1. Go to 2.

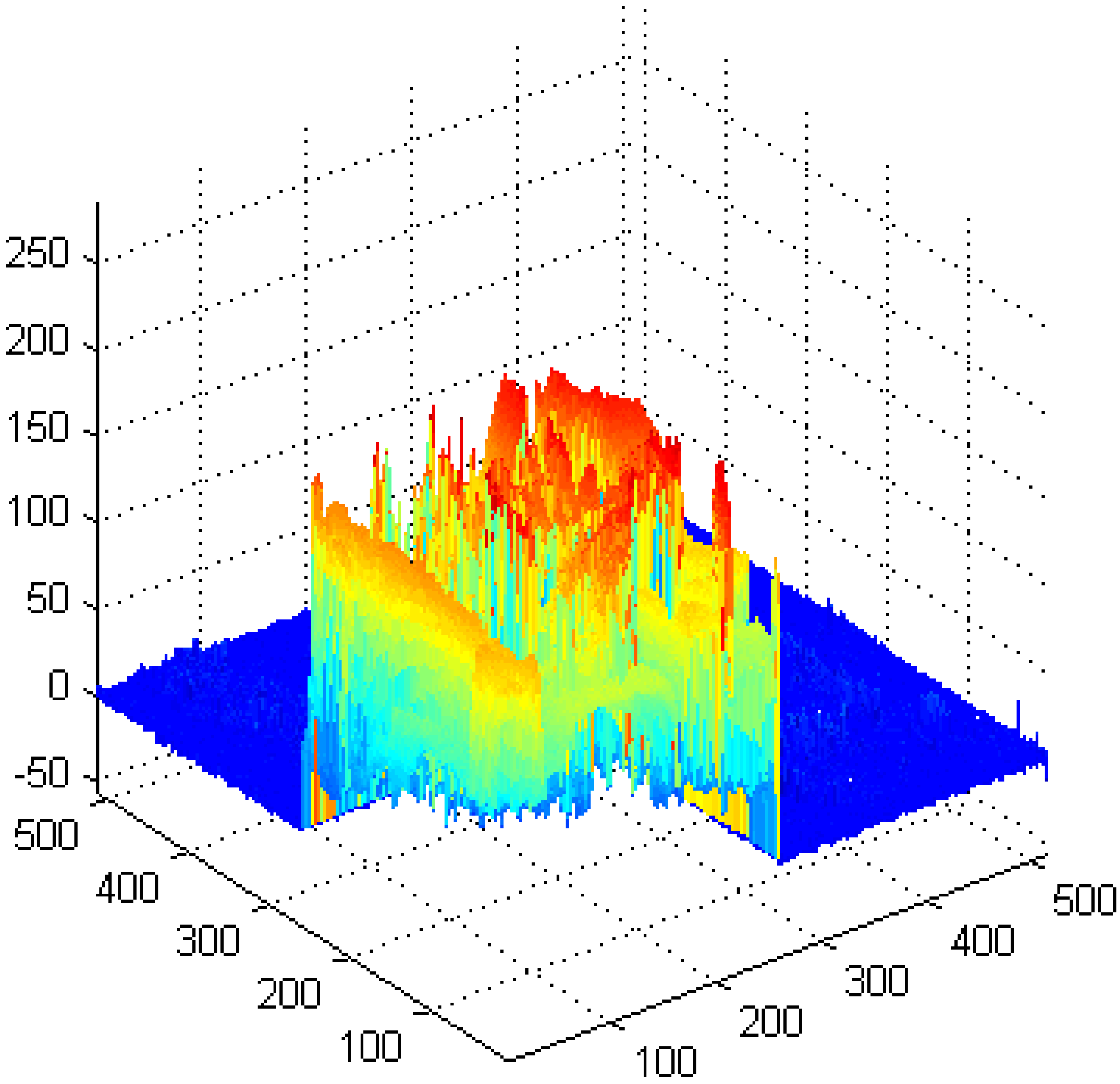
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500

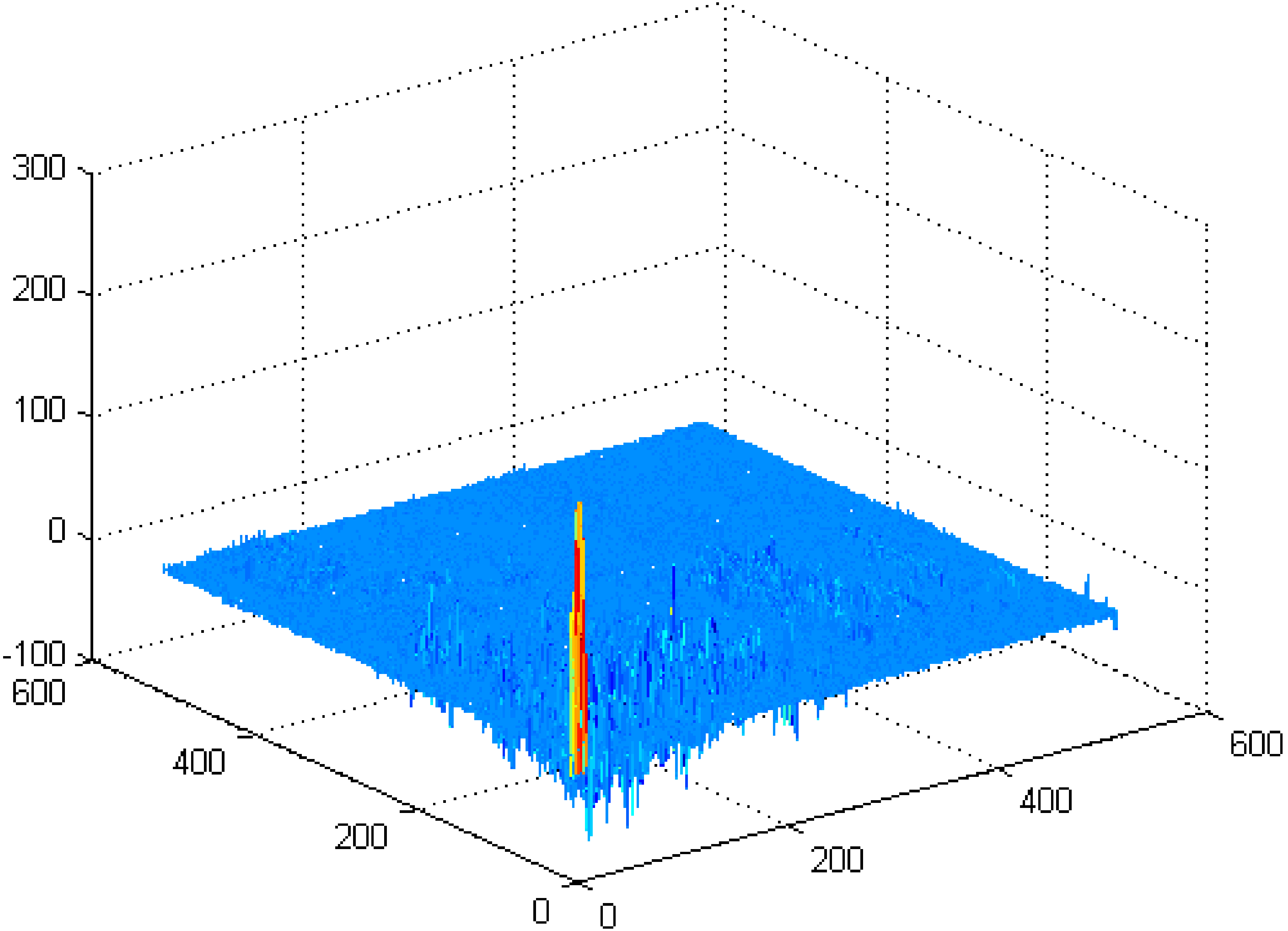


600
500
400
300
200
100
0
0
50
100
150
200
250

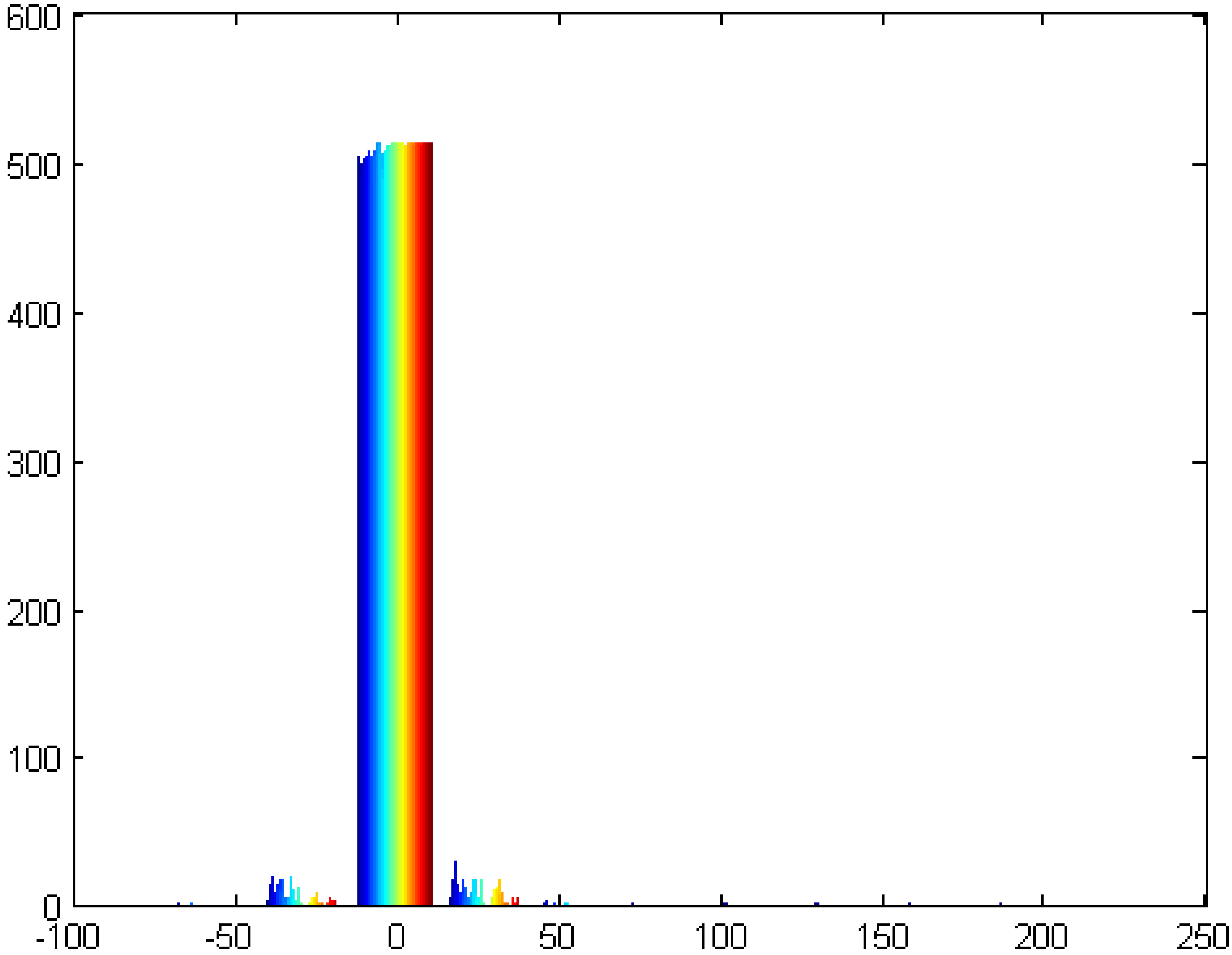


50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500


250
200
150
100
50
0
-50
500
400
300
200
100
100
200
300
400
500



300
200
100
0
-100
600
400
200
0
0
200
400
600



600
500
400
300
200
100
0
-100
-50
0
50
100
150
200
250


50
100
150
200
250
300
350
400
450
500
50 100 150 200 250 300 350 400 450 500

THE END