Wavelets and Images

Luiz Velho

IMPA
Instituto de Matematica Pura e Aplicada
Rio de Janeiro, Brazil

Outline

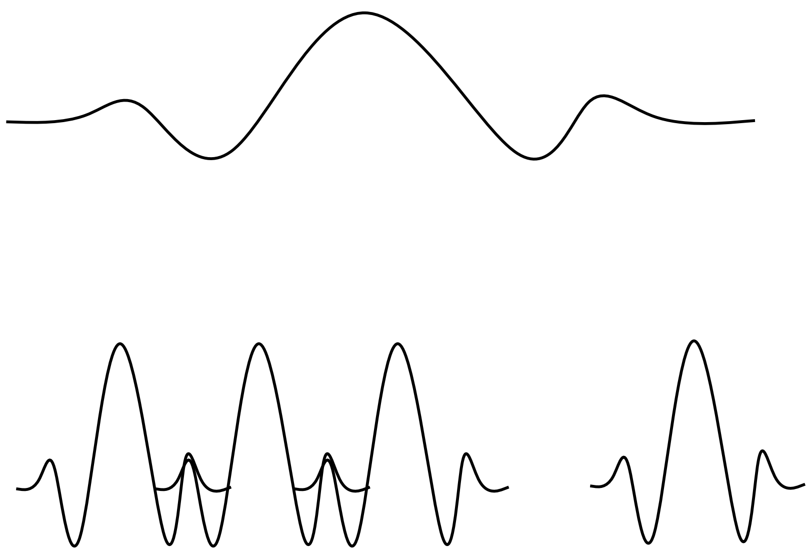
- Image Compression
- Image Querying
- Digital Painting
- Painting Tools

Image Compression

- Image Coherency
  - Spatial
  - Spectral
- Visual Information
  - Shade Gradations
  - Textures
  - Edges
- Compression Method
  - Eliminate Redundancy
  - Encode Information
- Types of Compression
  - Lossless
  - Lossy
- Problem: Decorrelate Data
  - Transform

Wavelets

- Wavelet Basis
  - Support Varies with Scale
  - Detect Transitions

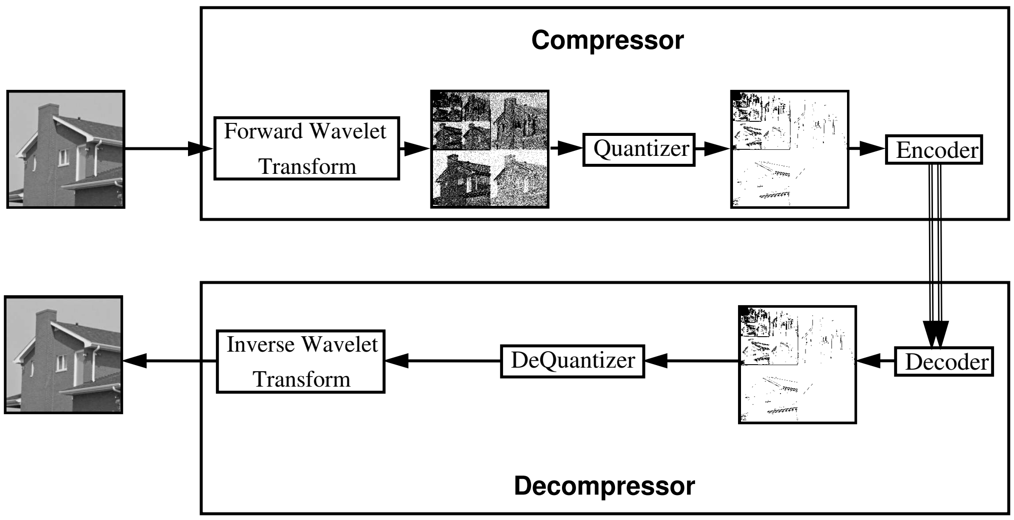


Properties of Wavelets for Compression

- Good Localization
  - Spatial and Frequency Domain
- Fast Computation
  - $\mathcal{O}(N)$
- Exact Decomposition / Reconstruction
  - Lossless Compression
- n Vanishing Moments
  - Higher Compression Rates
- Rational Coefficients
  - Hardware Implementation
- Subband Signal Decomposition
  - Flexible Framework

Basic Scheme

- Compression
  - FWT
  - Quantization
  - Encoder
- Decompression
  - Decoder
  - Dequantizer
  - IWT

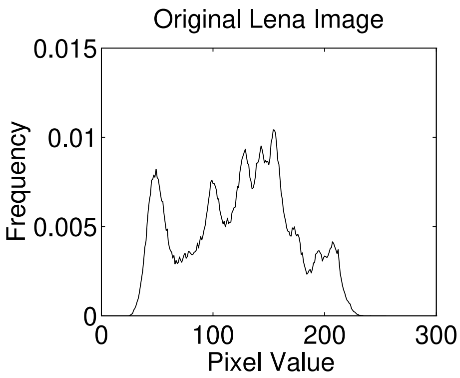


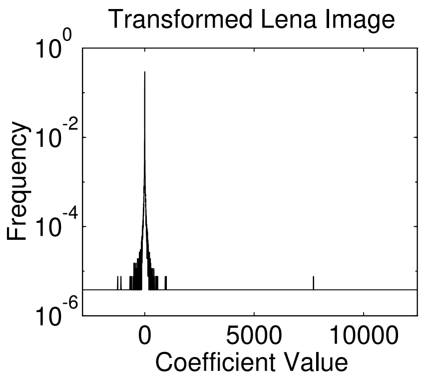
Wavelet Transform

- Entropy Coding
  - Orthogonal
  - Biorthogonal
- Vector Quantization
- Adaptive Transform
  - Wave Packets
- Edge Based
  - Derivative Dyadic Wavelets

Statistical Analysis

- Histogram





- Information Packing

- Lossless Compression
  - Lower Entropy

- Lossy Compression
  - Fewer Coefficients

Image Approximation

- Energy Invariance
  - Proportional Changes

- Mechanisms
  - Thresholding Function

    $$T(t,p) = \begin{cases} 0 & \text{if} |p| < t \\ p & \text{otherwise} \end{cases}$$
  - Quantization Function

    $Q(p)$
    - Quantization Cells
    - Quantization Levels

Quantization

- Optimal Quantization
  $$\min ||I(u,v) - \widetilde{I}(u,v)||$$
- Error Metric
  - Euclidean (RMS)
  - Perceptual
- Wavelets
  - Separate Quantizer for each Scale
- Quantizer Design
  - Properties of Visual System
    - Allocation of Bits
  - Statistics of Coefficients
    - Quantization Intervals

Encoding

- Dictionary of Symbols $\mathcal{D}$

- Types of Codes
  - Fixed Length
  - Variable Length
    - Huffman
    - Arithmetic

- Code Structure
  - Run-Length
  - Zero-Tree

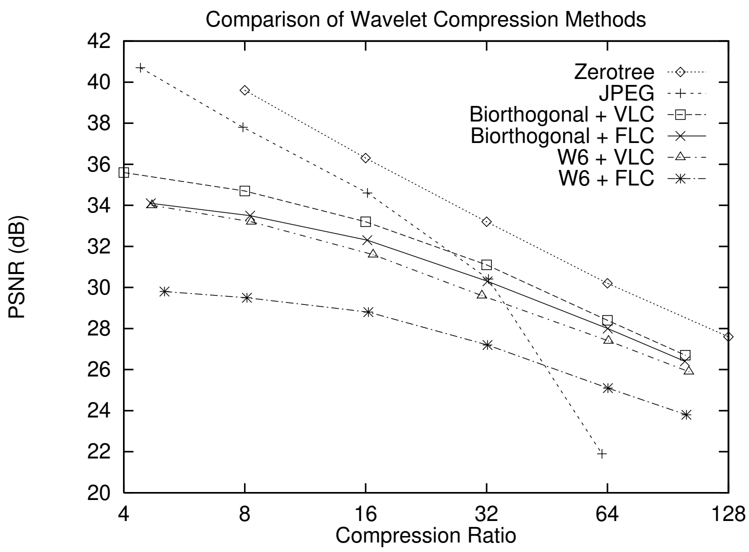
Example

- 32:1

- 128:1

Comparison with JPEG

- Image: Lena



- JPEG - up to 25:1

- Wavelet - 30:1 to 100:1

Image Querying

- Problem: Find an Image in a Database
- Types
  - Query by Content
    - Keywords
  - Query by Example
    - Similarities
- Image Match
  - Query Image: $Q[i, j]$
  - Target Image: $T[i, j]$

* *Distortions*

Query Metrics

- L_p Metrics (p = 1, 2)

$$||Q,T||_p = (\sum_{i,j} |Q[i,j] - T[i,j]|^p)^{1/p}$$

  - Expensive to Compute
  - Not Perceptual

- Color Histogram
  - Not Spatial
  - Color Shifts

Advantages of Wavelets

- Perceptual Metric
- Compression
  - Compact Signature
- Edge Detection
  - Key Features
- Multiscale
  - Resolution Independence of Q and T
- Fast, $\mathcal{O}(N)$
  - Real Time

Wavelet Metric

- Representation (M - wavelet transform)
  $M(0,0)$ Mean
  $M[i,j]$, $(i,j) \neq 0$: Wavelet Coefficients

- Metric

  $$||Q,T|| = w_{0,0}|Q[0,0] - T[0,0]| + \sum_{i,j} w_{i,j}|Q[i,j] - T[i,j]|$$

- Simplifying the Metric
  - Truncate coefficients: $T \in \mathbf{R} \rightarrow \widetilde{T} \in \{-1, 0, 1\}$
  - Substitute: $|Q[i,j] - T[i,j]|$ by $(Q[i,j] \neq T[i,j])$

- Rewriting for Fast Computation

  $\sum_{i,j}(Q[i,j] \neq T[i,j]) \rightarrow \sum_{i,j} w_k - \sum_{i,j}(Q[i,j] = T[i,j])$

- Final $||Q,T||$

  $$w_{0,0}|Q[0,0] - T[0,0]| - \sum_{i,j} w_{bin(i,j)}(Q[i,j] = T[i,j])$$

Tuning the Metric

$w_{i,j} = \text{bin}(i,j)$

- Adapt to a Class of Images : Training Sets
  - Scanned
  - Painted

- Solutions
  - Multidimensional Optimization
  - Logit Statistical Model

- Function

  $\text{bin}(i,j) = \mathsf{min}(\mathsf{max}(i,j), 5)$

- Weights

		Painted			*Scanned*	
b	w_Y	w_I	w_Q	w_Y	w_I	w_Q
0	4.04	15.14	22.62	5.00	19.21	34.37
1	0.78	0.92	0.40	0.83	1.26	0.36
2	0.46	0.53	0.63	1.01	0.44	0.45
3	0.42	0.26	0.25	0.52	0.53	0.14
4	0.41	0.14	0.15	0.47	0.28	0.18
5	0.32	0.07	0.38	0.30	0.14	0.27

Wavelet Transform

- Haar Wavelet

```
DecomposeImage(I) {
  for (row = 1 to r)
    DecompArray(I[row,v]);
  for (col= 1 to s)
    DecompArray(I[u,col]);
}

DecompArray(A[x])
{
  while (h>1) {
    h = h/2;
    for (i=0 to h-1) {
      B[i] = (A[2*i]+A[2*i+1])/sqrt(2);
      B[h+i] = (A[2*i]-A[2*i+])/sqrt(2);
    }
  }
  copy(B,A);
}
```

Database Structure

- Data
  - Values of $T[0,0]$
  - Signs of Largest m Coefs $T[i,j]$

- 6 Search Arrays
  (3 channels, pos/neg)

  $D^c_+[i,j]$

  $D^c_-[i,j]$
  - Entry is a List of Pointers

- Image Info
  - $T[0,0]$
  - Image Locator

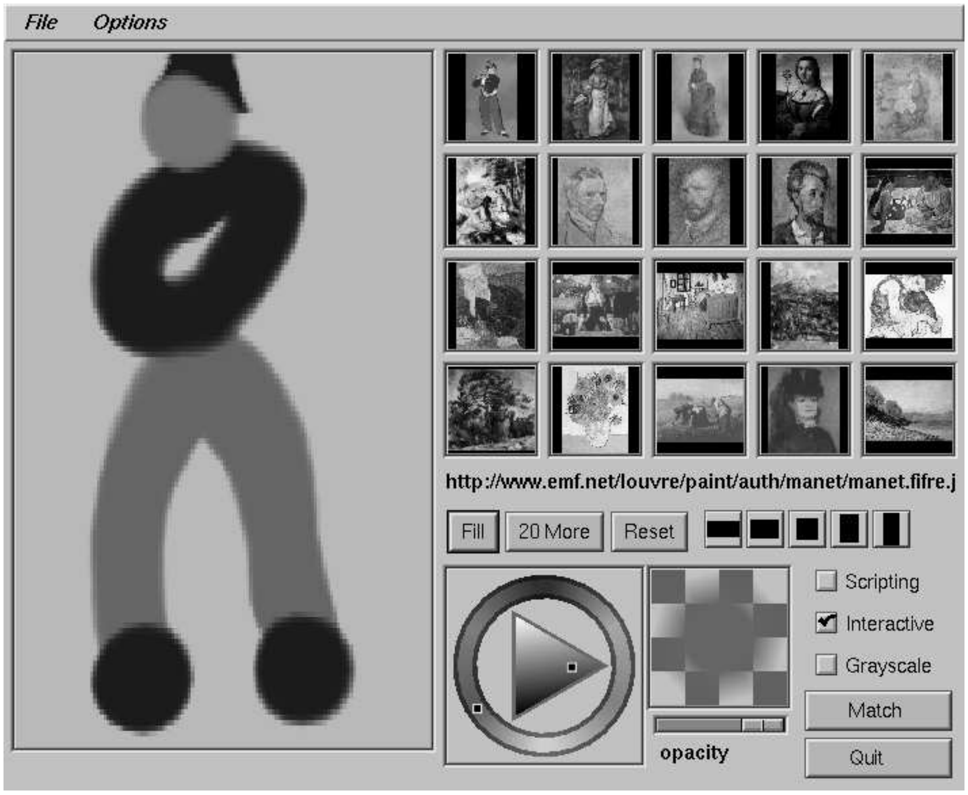
Computing a Score

```
ScoreQuery(Q, m)
{
  DecomposeImage(Q);

  InitScores(scores,0);

  foreach (color_channel c) {
    foreach (database_entry T)
      scores[index(T)]+=w[0]*|Q[0,0]-T[0,0]|;

    TruncateCoeffs(Q,m);

    foreach (Q[i,j] != 0) {
      if (Q[i,j] > 0)
        append(list, D_plus[i,j]);
      else
        append(list, D_minus[i,j]);

      foreach (element l of list)
        scores[index(l)] -= w[bin(i,j)]
    }
  }
  return scores
}
```

Finding a Match

- Rank all Images
  - Score Query
- Sort Scores
  - Heap Select
- Select Top 20
  - Show Thumbnail (5x4)
- Match
  - Visual Inspection

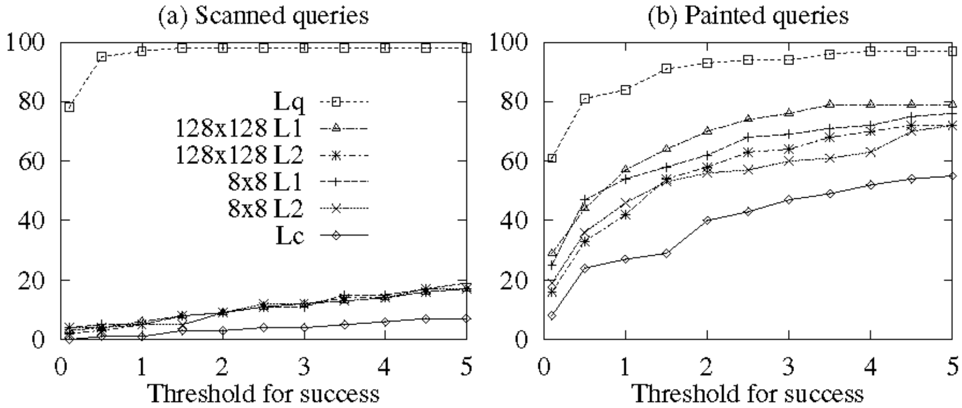
User Interface



- Query Type
  - Scan
    - File
  - Painted
    - Non-interactive
    - Interactive

Results

- Comparison



- Speed

$Metric$	$Time$	
	$n = 1093$	$n = 20,558$
Lq	0.19	0.44
$L^1(8 \times 8)$	0.66	7.46
$L^2(8 \times 8)$	0.68	6.39
$L^1(128 \times 128)$	47.46	892.60
$L^2(128 \times 128)$	42.04	790.80
Lc	0.47	5.03

* *Database Size*