Wavelet Transform for Image Processing

Zhang Jinrui[1]
He Jiashun
Meng Jingyuan
Jiang Zishen
Mo Zian

August 8, 2025

[1] alternative email:zhangjr1022@mails.jlu.edu.cn

1. Image Loading and Grayscale Conversion

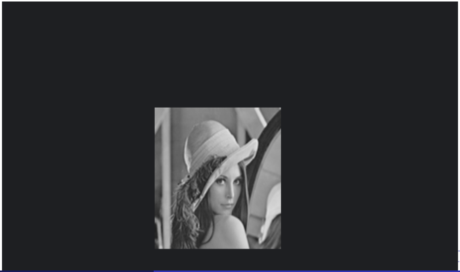
Step Explanation

The image is loaded using a Python imaging library and converted to grayscale.
This reduces data complexity and focuses analysis on structural content.

2. Rescaling and 3. Centered Cropping

Rescaling with Aspect Ratio Preservation

The image is resized using high-fidelity interpolation to ensure one side reaches the target length (2^N), without distortion.
The resized image is cropped to $2^N \times 2^N$, ensuring input uniformity without compromising important visual features.



4. Matrix Output

Step Explanation

The final image is converted into a matrix format suitable for:

- Mathematical operations (e.g., wavelet transforms)
- Storage and machine learning integration

128	135	142
130	138	145
125	132	139

Example of 3x3 image matrix (simplified)

5. Convert Matrix Information into an Image

Concept

Each matrix element represents the gray value of a pixel. Using this matrix, we can reconstruct the grayscale image.

```
[[162 161 162 ... 117 165 167]
 [160 160 160 ... 127 130 101]
 [157 156 157 ... 105  53  40]
 ...
 [ 54  56  58 ...  57  52  61]
 [ 50  53  52 ...  57  70  88]
 [ 48  53  49 ...  67  93 103]]
矩阵形状：(128, 128)
图像尺寸：(128, 128)
```

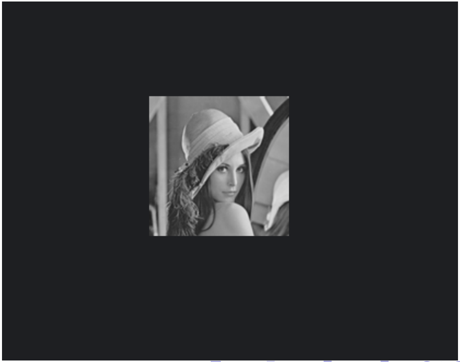


Figure: Matrix Information

Image compression

Let's see an application of our research first.

By our research, $\forall\ m \in \mathbb{N}_+$, we can get **an set of orthonormal bases** of $\mathbb{R}^m$ quickly.

Theorem 2.1

Let $\{u_i\}_{i=1}^m$ is a set of **orthonormal bases** of $\mathbb{R}^m$, $\{v_i\}_{i=1}^n$ is a set of **orthonormal bases** of $\mathbb{R}^n$, then

$$\{u_i \times v_j \mid i = 1, \cdots m, j = 1, \cdots n\}$$

is a set of **orthonormal bases** of $\mathbb{R}^{m \times n}$.

Image compression

A grayscale image is stored as a grayscale matrix in the computer. Then we only need to know how to "compress" a matrix.

For $A \in \mathbb{R}^{m\times n}$, if $\{u_i\}, \{v_j\}$ are sets of **orthonormal bases** of $\mathbb{R}^m$ and $\mathbb{R}^n$, according to (*Theorem 2.1*), $\exists\ \omega_{i,j} \in \mathbb{R}, i = 1, \cdots m, j = 1, \cdots, n$,

$$A = \sum_{i,j} \omega_{i,j} \ \cdot\ u_i \cdot v_j^T.$$

Let $M = (u_1, u_2, \cdots, u_m),\ N = (v_1, v_2, \cdots, v_n), W = (\omega_{i,j})$. Then

$$M^T AN = W,\ MWN^T = A.$$

Image compression

If $\omega_{i,j}$ is too small, the influence of corresponding basis can be ignored. Then

$$\widetilde{A} = \sum_{i,j} \widetilde{\omega_{i,j}} \cdot u_i \cdot v_j^T, \ \widetilde{\omega_{i,j}} = \begin{cases} \omega_{i,j}, & |\omega_{i,j}| > \lambda \\ 0, & |\omega_{i,j}| \leq \lambda \end{cases}.$$

$\widetilde{A}$ is an approximation of A, while λ is image compression strength.

Let $\widetilde{W} = (\widetilde{\omega_{i,j}})$. As long as λ is large enough, we can transform $\widetilde{W}$ into a **sparse matrix** which needs less storage space.

Then we set λ equal to 0.5 and see the result.

Image compression

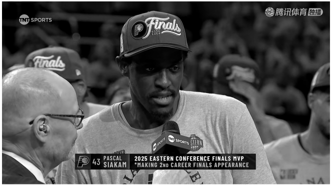


Figure: Original image

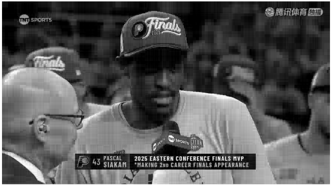


Figure: Final image

Image compression

For original image , the size of stored information is $981 \times 1759 = 1725579$.

However, after compression, the size of stored information is only $15364 \times 2 = 30728$.

Indeed, we reduce storage space by compression. However, we also lose much detailed information of original picture. That's why we should change λ for different purposes to reach the balance.

Research background and objectives

Advantages of Wavelet Transform

- Multiresolution analysis
- frequency localization capability

Importance of Energy Conservation

- Ensures information integrity in signal processing
- Theoretical basis for image compression, denoising

Research Objectives

- Validate energy conservation mathematically
- Quantify energy retention via image compression experiments

Theoretical analysis

Theoretical Foundations of DWT

- Scaling function (ϕ) and wavelet function (ψ)
- Multi-scale decomposition: Approximation (cA) and Detail (cH, cV, cD) subbands
- Orthogonal Wavelet Basis Properties:

$$\langle \psi_{j,k}, \psi_{m,n} \rangle = \delta_{j,m} \delta_{k,n}, \qquad j, k, m, n \in \mathbb{Z}$$

Critical Conditions

- Use of orthogonal wavelets (e.g., Haar, Db1)
- No signal truncation or padding-induced errors

Energy Conservation Formula

Energy Conservation Formula for Orthogonal DWT

- For orthogonal wavelet bases:

$$\sum_{i=1}^{M}\sum_{j=1}^{N}|x[i,j]|^2 = \sum_{k,l}|d_k[l]|^2 + \sum_{m}|a_J[m]|^2$$

- For Haar wavelet bases:

$$\sum_{i,j}|x[i,j]|^2 = \sum_{i,j}|cA[i,j]|^2 + \sum_{i,j}|cH[i,j]|^2 + \sum_{i,j}|cV[i,j]|^2 + \sum_{i,j}|cD[i,j]|^2$$

Experimental Design and Code Implementation

Workflow

- Image preprocessing
  - grayscale conversion
  - resizing to power-of-2 dimensions
- DWT decomposition("dwt2" function)
- Energy calculation and conservation validation
- Image reconstruction("idwt2" function)

Parameters

- Test image:"Lena"(52*52)
- Wavelet basis:Haar

Code and Implementation

matlab

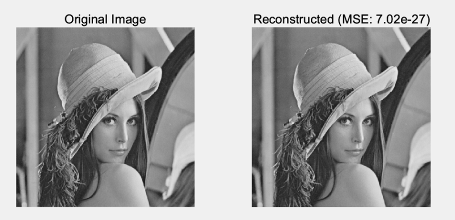
```matlab
[cA, cH, cV, cD] = dwt2(A, 'haar');
energy_transformed = sum(cA(:).^2) +
sum(cH(:).^2) + sum(cV(:).^2) +
sum(cD(:).^2);
```

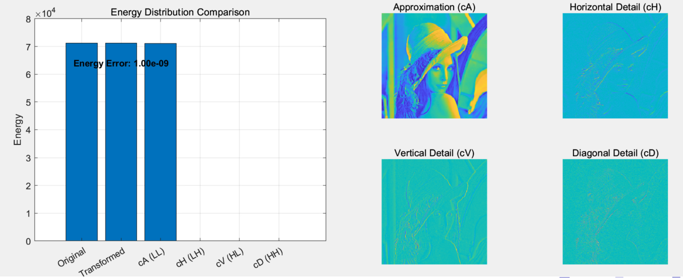
Figure: "Haar" basis

```matlab
%% 5. Verify Energy Conservation
energy_error = abs(energy_original - energy_transformed);
fprintf('Verify Energy Conservation:\n');
fprintf(' |E_original - E_transformed| = %.8e\n', energy_error);
if energy_error < 1e-8
    fprintf('Energy Conservation holds\n');
else
    fprintf('Energy Conservation violated\n');
end

```

Code and Implementation





Basic Idea

- $$\sum_{k=1}^{n} \hat{I}_k^4 \geq (\sum_{k=1}^{n} \hat{I}_k^2)^2 = (\sum_{k=1}^{n} I_k^2)^2$$

- $$\min_{\hat{I}_k \in l^2 st. \sum_{k=1}^{n} \hat{I}_k^2 = \sum_{k=1}^{n} I_k^2} (-\sum_{k=1}^{n} \hat{I}_k^4)$$

Basic Idea

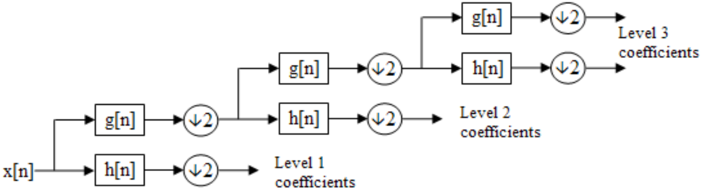
- 

  Figure: Cascading

- $$\min_{\Phi}(-\sum_{k=1}^{n} \hat{l}_k^4)$$

- Condense the energy to as less coefficients as possible.

Results

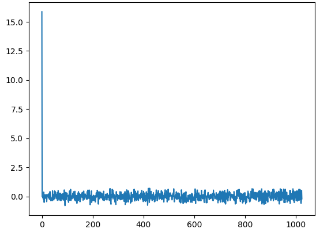
- 

Figure: Haar 2x2 filter bank random input. Compression Rate 0.2568

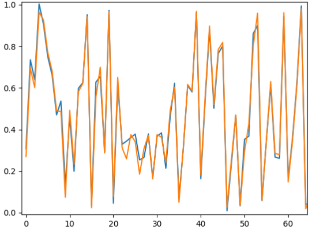


Figure: Haar 2x2 filter bank random input.Total average energy loss 0.0009

Results

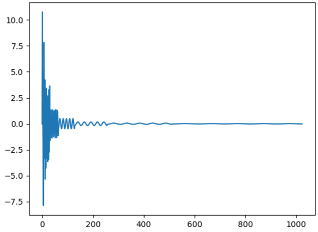
- 

Figure: Haar 2x2 filter bank sin input frequency. Compression Rate 0.9287

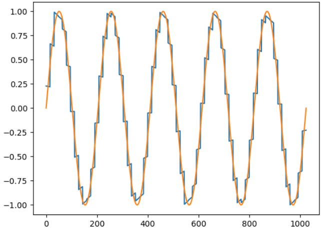


Figure: Haar 2x2 filter bank sin input reconstruction.Total average energy loss 0.0104

Results

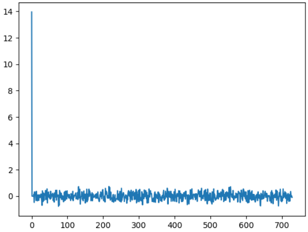
- 

Figure: Haar 3x3 filter bank random input. Compression Rate 0.1906

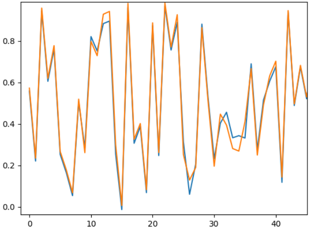


Figure: Haar 3x3 filter bank random input.Total average energy loss 0.0008

Results

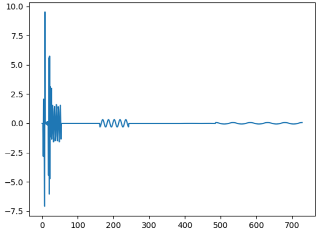


Figure: Haar 3x3 filter bank sin input frequency. Compression Rate 0.7750

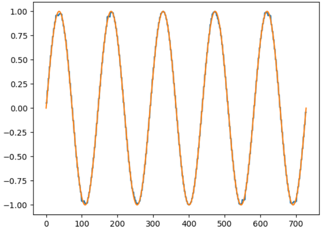


Figure: Haar 3x3 filter bank sin input reconstruction.Total average energy loss 0.0007

Sequence Spaces

- $\ell(\mathbb{Z}) = \left\{v = \{v(k)\}_{k\in\mathbb{Z}} \mid v(k) \in \mathbb{C}\right\}$
- $\ell_0(\mathbb{Z}) = \left\{u \in \ell(\mathbb{Z}) \mid \text{only finitely many } u(k) \neq 0\right\}$
- filter bank: $\{u_0, \ldots, u_s\} \subset \ell_0(\mathbb{Z})$

Operators

- **Subdivision:**

$$[S_u v](n) = 2 \sum_{k \in \mathbb{Z}} v(k)\, u(n-2k)$$

- **Transition:**

$$[T_u v](n) = 2 \sum_{k \in \mathbb{Z}} v(k)\, \overline{u(k-2n)}$$

- **Discrete Framelet Transform (DFrT):**

$$\frac{1}{2} \sum_{\ell=0}^{s} S_{u_\ell} T_{\tilde{u}_\ell} v$$

Perfect Reconstruction Condition

- the transform satisfies perfect reconstruction if

$$\frac{1}{2}\sum_{\ell=0}^{s} S_{u_\ell} T_{\tilde{u}_\ell} v = v \quad \forall v \in \ell(\mathbb{Z}).$$

Main Theorem – Four Equivalent Statements

The following are equivalent:

1. The filter bank $(\{\tilde{u}_\ell\}, \{u_\ell\})$ has PR.
2. The identity $\frac{1}{2}\sum_{\ell=0}^{s} S_{u_\ell} T_{\tilde{u}_\ell} v = v$ holds for all $v \in \ell_0(\mathbb{Z})$.
3. The identity $\frac{1}{2}\sum_{\ell=0}^{s} S_{u_\ell} T_{\tilde{u}_\ell} v = v$ holds for $v = \delta$ and $v = \delta(\cdot - 1)$.
4. Frequency-domain identities for all $\omega \in \mathbb{R}$:

$$\sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega)}\, \hat{u}_\ell(\omega) = 1,$$

$$\sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega)}\, \hat{u}_\ell(\omega + \pi) = 0.$$

(i) $\Rightarrow$ (ii) $\Rightarrow$ (iii)

Trivial inclusions:

$$\ell_0(\mathbb{Z}) \subseteq \ell(\mathbb{Z}) \quad \text{and} \quad \delta, \delta(\cdot - 1) \in \ell_0(\mathbb{Z}).$$

(iii) ⇒ (iv)

1. **Frequency Domain Conversion:** Using Fourier transforms of $T_{\tilde{u}_\ell}$ and S_{u_ℓ}:

$$\widehat{T_{\tilde{u}_\ell} v}(\omega) = \hat{v}(\omega/2)\overline{\hat{\tilde{u}}_\ell(\omega/2)} + \hat{v}(\omega/2+\pi)\overline{\hat{\tilde{u}}_\ell(\omega/2+\pi)},$$

$$\widehat{S_{u_\ell} w}(\omega) = \hat{w}(2\omega)\hat{u}_\ell(\omega).$$

2. **Substitute Basis Signals:** Plugging $v = \delta$ ($\hat{v} = 1$) and $v = \delta(\cdot - 1)$ ($\hat{v} = e^{-i\omega}$) into the PR condition yields the system:

$$\sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega)}\hat{u}_\ell(\omega) + \sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega+\pi)}\hat{u}_\ell(\omega+\pi) = 1,$$

$$\sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega)}\hat{u}_\ell(\omega) - \sum_{\ell=0}^{s} \overline{\hat{\tilde{u}}_\ell(\omega+\pi)}\hat{u}_\ell(\omega+\pi) = 1.$$

3. **Solve System:** Adding and subtracting these equations gives condition (iv).

(iv) $\Rightarrow$ (ii) and (ii) $\Rightarrow$ (i)

- (iv) $\Rightarrow$ (ii): Condition (iv) implies the frequency identity holds for all $v \in \ell_0(\mathbb{Z})$ by linearity of Fourier transform.
- (ii) $\Rightarrow$ (i): Localization Argument: For any $v \in \ell(\mathbb{Z})$, truncate to local signal $v_n \in \ell_0(\mathbb{Z})$ using finite support of filters. Apply (ii) to show:

$$v(n) = \frac{1}{2}\sum_{\ell=0}^{s}[S_{u_\ell} T_{\tilde{u}_\ell} v](n).$$