Wavelet Transform in Practice

From Theory to Production-Ready Python **Applications**

VOLUME I

Foundations and **Core Wavelet Theory**

Concepts, Mathematics, and Python Foundations

Shouke Wei

Wavelet Transform in Practice

From Theory to Production-Ready Python Applications

Series

Wavelet Transform in Practice

From Theory to Production-Ready Python Applications

VOLUME I

Foundations and Core Wavelet Theory

Concepts, Mathematics, and Python Foundations

Shouke Wei

DEEPSIM
PRESS

First Edition, 2026

ISBN 978-1-0699284-1-2 (eBook)
DOI 10.5281/zenodo.17373653

Published by
Deepsim Press
An independent imprint of Deepsim Intelligent Technology Inc.
Abbotsford, British Columbia, Canada
https://press.deepsim.ca

This book is intended for educational and professional readers.

For resources, updates, and companion code, visit:
`https://press.deepsim.ca/wavelet-books`

DEEPSIM
PRESS

About the Author

Shouke Wei, Ph.D., is a researcher, scientist, and entrepreneur specializing in intelligent IoT systems, robotics, big data analytics, modeling and forecasting, early-warning systems, and edge computing. With academic and industry experience across Europe, North America, and Asia, Dr. Wei is recognized for bridging advanced theory with real-world, production-ready systems.

Dr. Wei earned his Ph.D. in Environmental and Resource Management from the Department of Environmental Informatics at Brandenburg University of Technology Cottbus–Senftenberg (Germany). He conducted postdoctoral research at the Swiss Federal Institute of Aquatic Science and Technology (Eawag), where he also served as a doctoral supervisor, and held research positions at the University of British Columbia (Canada).

Recognized as a National High-End Talent (Class A) in China, Dr. Wei has held distinguished and adjunct professorships at multiple institutions, including Yantai University, Ludong University, and Jining University. He has served as a graduate supervisor and distinguished professor in computer science, control engineering, and applied mathematics.

Dr. Wei currently serves as CEO and Chief Scientist of Deepsim Intelligent Technology Inc. (Canada), Chief Scientist at Canadian Sincerity Enterprises Inc., and Chief Scientist of Shandong Deepsim Intelligent Technology Co., Ltd. He is also a Postdoctoral Co-Supervisor at the Shandong Postdoctoral Innovation Practice Base and currently serves as Director of Qilu Artificial Intelligence and Digital Manufacturing Innovation at Shandong Deepsim Intelligent Technology Co., Ltd., China.

Dr. Wei has led or contributed to 19 major international research projects and the development of 19 intelligent systems, including autonomous water-quality monitoring vessels, AI-based environmental early-warning platforms, medical image diagnosis models, precision agriculture robots, and autonomous service robots.

His scholarly contributions include 40+ peer-reviewed publications and 500+ tutorial online articles, 6 patents, 30 software copyrights, and 2 China national scientific and technological achievements. Dr. Wei's work focuses on making advanced computational methods—particularly wavelet-based signal processing—accessible, practical, and impactful for researchers and practitioners worldwide.

For more info, visit: `https://shouke.deepsim.ca`

Contents

Preface

Wavelet transforms have become fundamental tools in modern signal processing, image analysis, time-series modeling, and data security. Their mathematical foundations are well established in the literature; however, the practical implementation of wavelet methods—particularly within contemporary computational environments such as Python—often presents significant challenges. This three-volume series, *Wavelet Transform in Practice: From Theory to Production-Ready Python Applications*, was conceived to bridge the gap between theoretical rigor and reproducible, real-world implementation across diverse application domains.

Rather than emphasizing mathematical abstraction alone, the series integrates conceptual foundations, algorithmic structure, and carefully constructed Python implementations. The objective is to provide readers not only with theoretical understanding, but also with the methodological clarity necessary to apply wavelet transforms effectively in research and production environments.

This first volume, *Foundations and Core Wavelet Theory: Concepts, Mathematics, and Python Foundations*, establishes the theoretical and computational groundwork for the volumes that follow. It introduces the fundamental principles of wavelet analysis, including core mathematical constructs and their relationship to classical signal-processing methods. Building upon this foundation, the volume presents essential discrete wavelet techniques—implemented using the PyWavelets library—while maintaining a consistent emphasis on clarity, structure, and reproducibility.

Who This Book Is For

This volume is intended for:

- Data scientists and machine learning engineers seeking principled multiscale feature extraction techniques
- Signal processing practitioners working with biomedical, audio, or sensor data
- Research scientists in environmental monitoring, geophysics, finance, and related fields
- Graduate students requiring both theoretical grounding and practical implementation guidance
- Software engineers integrating wavelet-based analysis into analytical pipelines or production systems

Readers interested in denoising signals, compressing images, detecting anomalies in time series, or constructing multiscale representations for machine learning will find in this volume both conceptual structure and practical tools.

Organization of This Volume

The book is organized into three parts, guiding the reader from foundational theory to structured implementation:

Part I: Foundations of Wavelet Theory and Computational Frameworks
Introduces the conceptual and mathematical foundations of wavelet analysis, including scaling functions, orthonormal bases, and multiresolution structure. It also establishes the computational framework through an introduction to PyWavelets, ensuring that theoretical constructs are immediately connected to practical implementation.

Part II: Core Techniques and Implementations
Presents one-dimensional, two-dimensional, and n-dimensional Discrete Wavelet Transforms (DWT), with attention to algorithmic formulation, visualization, and structured Python workflows.

Part III: Multiresolution and Extended Discrete Wavelet Methods
Examines advanced multiresolution frameworks, including the Stationary Wavelet Transform (SWT), Multiresolution Analysis (MRA), and Wavelet Packet Transforms (WPT), emphasizing both theoretical coherence and implementation detail.

Prerequisites and Computational Environment

This volume assumes:

- Working knowledge of Python programming
- Familiarity with NumPy and Matplotlib
- Basic exposure to signal processing concepts (helpful but not strictly required)
- General awareness of machine learning concepts for advanced sections

The primary computational tools include:

- **PyWavelets** for discrete wavelet analysis
- **NumPy, SciPy, and Matplotlib** for numerical computation and visualization
- **scikit-image (skimage)** for image processing and multidimensional signal analysis
- **imageio** for image input/output and data handling
- Additional libraries introduced as required for specific analytical tasks

On Code and Reproducibility

All code examples are designed to be self-contained and reproducible. When real-world datasets are referenced, clear instructions for access are provided. Where proprietary data cannot be shared, synthetic datasets are constructed to preserve essential structural characteristics.

Implementation examples prioritize conceptual clarity and methodological correctness. Computational optimization strategies are discussed where relevant, but the primary goal is to illuminate structure rather than to pursue maximal efficiency.

Looking Forward

Wavelet analysis continues to evolve, with increasing relevance in hybrid machine learning systems, interpretable AI, edge computing, and large-scale scientific data analysis. As readers progress through this volume, they are encouraged to consider how multiscale representations may illuminate structure within their own domains of inquiry.

It is my hope that this volume serves not only as a technical reference, but also as a foundation for continued exploration and innovation in wavelet-based analysis. The subsequent volumes extend these foundations into applied methodologies and domain-specific systems, completing the progression from theoretical principles to production-ready implementations.

Shouke Wei, PhD
Deepsim Intelligent Technology Inc.
Deepsim Academy
Abbotsford, Canada
January 17, 2026

Acknowledgments

Writing this book has been an exciting and deeply rewarding journey—one that weaves together years of academic research, software development, and teaching. I am profoundly grateful to all those who have supported, challenged, and inspired me along the way.

I extend my heartfelt thanks to my colleagues and collaborators in both academia and industry for their insights, encouragement, and stimulating discussions, particularly on projects involving wavelets and real-time systems. Their perspectives and expertise have significantly shaped the ideas and applications presented in this book. To my students and research assistants, thank you for your curiosity, your thoughtful questions, and for continually challenging me to explain complex concepts with greater clarity and precision. Teaching and learning alongside you has been a constant source of motivation.

I am especially grateful to my family, whose unwavering support, patience, and encouragement sustained me through every stage of the writing process. This book is as much yours as it is mine.

Finally, this work stands on the shoulders of giants. It draws upon decades of pioneering research in wavelet theory by visionaries such as Ingrid Daubechies, Stéphane Mallat, and many others who laid the mathematical and practical foundations of modern wavelet analysis. My sincere appreciation also goes to the developers and contributors of PyWavelets, SciPy, NumPy, Matplotlib, skimage, imageio and the broader open-source community. Your dedication and collaborative spirit have made it possible to build reproducible, robust, and elegant tools for wavelet, signal, and time series analysis in Python, bringing the power of modern scientific computing to researchers, educators, and practitioners worldwide.

Beyond the technical foundations, this book was written using Python, Jupyter Notebook/Lab, Jupyter Book, Quarto, MyST Markdown, and LaTeX. I am grateful to the communities behind these tools for enabling clear, structured, and high-quality scholarly communication.

Without these people, libraries, packages, and tools, this book would not have been possible. Thank you all.

Lists of Symbols

Symbol	Meaning	Description
Wavelet Functions		
$\psi(t)$	Mother Wavelet	Oscillatory, zero-mean function for analyzing high-frequency components
$\varphi(t)$	Scaling Function	Function representing low-frequency approximation components
$\psi_{j,k}(t)$	Discrete Wavelet	$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k)$, scaled and shifted wavelet
$\psi_{a,b}(t)$	Continuous Wavelet	$\psi_{a,b}(t) = \frac{1}{\sqrt{\lVert a \rVert}}\psi(\frac{t-b}{a})$, with scale a and shift b
Parameters		
a	Scale Parameter	Controls dilation of wavelet, $a \in \mathbb{R}^+$
b	Shift Parameter	Controls position of wavelet in time, $b \in \mathbb{R}$
j	Scale Level	Discrete scale index; higher $j =$ coarser resolution
k	Translation Index	Discrete position index
t	Time Variable	Continuous time parameter
n	Discrete Index	Integer index for discrete signals
Signals		
$x(t)$	Continuous Signal	Original continuous signal being analyzed

Symbol	**Meaning**	**Description**
$x[n]$	Discrete Signal	Discrete input signal at sample index n
$\hat{x}(t)$	Reconstructed Signal	Signal reconstructed from wavelet coefficients
$\hat{x}(\omega)$	Fourier Transform	Frequency-domain representation of signal
Coefficients		
$W(a,b)$	CWT Coefficients	Continuous wavelet transform coefficients
A_j or cA_j	Approximation	Low-frequency coefficients at level j
D_j or cD_j	Detail	High-frequency coefficients at level j
Filters		
$h[n]$	Low-Pass Filter	Decomposition filter for extracting approximation
$g[n]$	High-Pass Filter	Decomposition filter for extracting detail
$\tilde{h}[n]$	Reconstruction Low-Pass	Synthesis filter for recovering approximation
$\tilde{g}[n]$	Reconstruction High-Pass	Synthesis filter for recovering detail
$\downarrow 2$	Downsampling	Decimation: keeps every second sample
$\uparrow 2$	Upsampling	Inserts zeros between samples
2D Transform		
$I(x,y)$	2D Image	Two-dimensional signal or image
LL_j	Approximation Subband	Low-pass in both directions
LH_j	Horizontal Detail	Low-pass rows, high-pass columns
HL_j	Vertical Detail	High-pass rows, low-pass columns
HH_j	Diagonal Detail	High-pass in both directions
Decomposition		
J	Maximum Level	Deepest level of decomposition
2^j	Scale Factor	Resolution at level j

Symbol	Meaning	Description
Operations		
$\langle f, g \rangle$	Inner Product	$\int f(t)\overline{g(t)}\,dt$
$\|\cdot\|$	Norm	Magnitude or length
$*$	Convolution	$(f * g)(t) = \int f(\tau)g(t-\tau)\,d\tau$
E	Energy	Sum of squared coefficients: $E = \sum \|c\|^2$
Frequency		
ω	Angular Frequency	Frequency in radians per second
f	Frequency	Frequency in Hertz (Hz)
f_s	Sampling Frequency	Sampling rate of discrete signal
$\hat{\psi}(\omega)$	Wavelet Spectrum	Fourier transform of wavelet function
Thresholding		
λ	Threshold Value	Coefficient threshold for denoising
$T_h(\cdot, \lambda)$	Hard Thresholding	Sets coefficients below λ to zero
$T_s(\cdot, \lambda)$	Soft Thresholding	Shrinks coefficients toward zero
σ	Noise Level	Standard deviation of noise
Mathematical Sets		
$\mathbb{R}$	Real Numbers	Set of all real numbers
$\mathbb{R}^+$	Positive Reals	Positive real numbers only
$\mathbb{Z}$	Integers	Integer numbers
$L^2(\mathbb{R})$	Square-Integrable	Functions with finite energy
Wavelet Families		
Haar	`'haar'`	Simplest wavelet; discontinuous
Daubechies	`'db4'`, `'db8'`	Compactly supported orthogonal wavelets
Symlet	`'sym4'`, `'sym8'`	Near-symmetric wavelets
Coiflet	`'coif2'`	Wavelets with vanishing moments
Properties		

Symbol	**Meaning**	**Description**
N	Vanishing Moments	$\int t^k \psi(t)\, dt = 0$ for $k < N$
N_h	Filter Length	Number of coefficients in filter
Statistics		
μ	Mean	Average value
σ^2	Variance	Measure of spread
SNR	Signal-to-Noise Ratio	Ratio of signal to noise power
MSE	Mean Squared Error	$\frac{1}{N} \sum (x_i - \hat{x}_i)^2$

Setup Python Environment

This preliminary chapter outlines **two recommended methods** for setting up your Python environment: **uv**, a modern, fast, and isolated tool; and **Anaconda**, a traditional yet comprehensive distribution.

Create Your Project Folder

The easiest way is to visit press.deepsim.ca/wavelet-books/vol-1, or github.com/shoukewei/wavelet-books/vol-1 to download the full project.

However, you can creat your project folder manually or use the command, and then navigate into it using the command:

```
mkdir wavelet-books
cd wavelet-books
```

Recommended folder structure:

```
wavelet-books/
   notebooks/           # All Jupyter notebooks
   data/                # Example datasets / signals
   scripts/             # Python scripts (optional)
   requirements.txt     # Dependencies
   pyproject.toml       # Optional modern environment file
   uv.lock              # File pinning exact packages versions
   README.md
```

Then you just download the `requirements.txt`, `pyproject.toml` and `uv.lock` files and the data you need.

Option 1: Using `uv` (Recommended)

`uv` is a modern Python package manager that offers instant environments, conflict-free dependency management, and fast reproducibility.

1. Install `uv`

```
pip install uv
# or
curl -LsSf https://astral.sh/uv/install.sh | sh
```

2. Create and activate a virtual environment

Go to your project folder and run the following command to create a virtual environment and activate it:

```
uv venv
source .venv/bin/activate  # On macOS/Linux
.venv\Scripts\activate    # On Windows
```

3. Install dependencies

Choose one of the following two ways:

(1) Install all dependencies defined in `pyproject.toml` (Recommand)

```
uv sync
```

This will:

- Create a virtual environment (if none exists).
- Install all dependencies (and optional dev dependencies, if specified).
- Lock versions into a uv.lock file for reproducibility.

(2) Install all dependencie defined in `requirements.txt`

```
uv pip install -r requirements.txt
```

Or:

```
uv pip install -r requirements.txt
```

If you need to install additional packages, use the following command:

```
uv add [package-name]
```

4. Launch Jupyter Lab

You can launch Jupyter Lab using the following command:

```
uv run jupyter lab
```

Or Jupyter Notebook:

```
uv run jupyter notebook
```

Now you're ready to explore wavelet analysis interactively!

Option 2: Using Anaconda (Alternative)

Anaconda is a popular Python distribution that bundles scientific computing libraries and tools.

1. Install Anaconda

Download and install from https://www.anaconda.com/download

2. Create a new environment

```
conda create -n wavelet python=3.11
conda activate wavelet
```

3. Install dependencies

Go to your project folder and run the following one of the commands:

(1) Install all dependencies defined in `pyproject.toml` (Recommand)

(2) Install all dependencie defined in `requirements.txt`

```
conda install --file requirements.txt
```

Or use pip to install all dependencies:

```
pip install -r requirements.txt
```

(3) Install required packages (individually or more). For example:

```
conda install pywavelets numpy matplotlib pandas scipy
 ↪  jupyterlab
```

4. Launch Jupyter Notebook

```
jupyter lab
```

Verify Installation

Run the following to confirm everything works:

```
import pywt, numpy as np, matplotlib.pyplot as plt
print("PyWavelets version:", pywt.__version__)
```

You should see something like:

```
PyWavelets version: 1.8.0
```

If you get any errors, please check your Python version and ensure all dependencies are correctly installed.

Part I

Foundations of Wavelet Theory and Computational Frameworks

1 Introduction to Wavelets

The wavelet transform has emerged as one of the most powerful tools in signal processing, data compression, and time-frequency analysis. Unlike Fourier transform (FT), which provides only frequency information, It offers both time and frequency localization, making it especially suitable for non-stationary signals (S. Mallat 1989).

The origin of wavelet theory can be traced back to the study of functional spaces and harmonic analysis. The modern development of wavelets began in the 1980s, particularly through the pioneering works of Ingrid Daubechies, who introduced a family of orthonormal wavelets with compact support (Daubechies 1992).

In this book, it focuses on both the theoretical foundation and practical application of wavelet transforms using Python, specifically the PyWavelets. PyWavelets is an open-source Python library that provides a wide range of wavelet transforms, filter design, and utility functions that facilitate wavelet-based analysis.

This chapter sets the stage for deeper exploration in subsequent chapters. It will review the historical context, highlight key properties of wavelets, and discuss the motivations behind adopting the wavelet transform in real-world scenarios. For a deeper understanding of wavelets and operators, Meyer's foundational text is highly recommended (Meyer 1993).

1.1 Overview

This chapter introduces the fundamental concepts of wavelets and wavelet transforms, including:

- **Core Principles and Significance:** Explains the essential principles of wavelets and wavelet transforms and their importance in signal and data analysis.

- **Advantages over Traditional Methods:** Discusses why wavelet transforms are powerful tools for analyzing non-stationary signals or time-varying signals, particularly in cases where conventional techniques such as the Fourier transform are less effective.

- **Comparative Analysis:** Provides a detailed comparison between Wavelet transforms and Fourier transforms, emphasizing their respective strengths, limitations, and areas of application.

- **Classification of Wavelet Transforms:** Outlines the main categories of wavelet transforms from multiple perspectives, including dimensionality, decomposition levels, and types of discrete wavelet transforms.

This foundational overview sets the stage for deeper mathematical and practical exploration of wavelet theory in the subsequent chapters.

1.2 What Are Wavelets?

A wavelet is a localized oscillatory function with finite energy, used to analyze signals at multiple scales and positions in time or space. Wavelets are classified into different families based on their mathematical properties. Common wavelet families include Haar, Daubechies, Symlets, Coiflets, Morlet, Mexican Hat, Complex Shannon, and Meyer wavelets (Figure 1.1), each providing distinct trade-offs between time localization, frequency resolution, and computational efficiency.

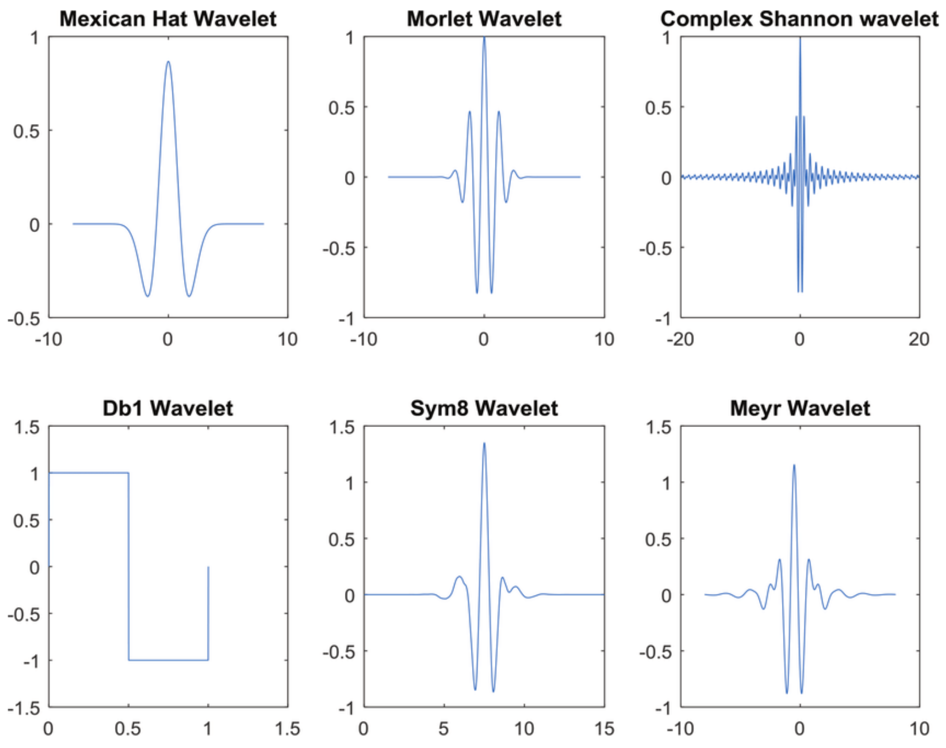


Figure 1.1: Examples of commonly used wavelets, illustrating their diverse shapes and properties. The Mexican Hat and Morlet wavelets are continuous wavelets with good time–frequency localization; the Complex Shannon and Meyer wavelets provide sharp frequency resolution; while the db1 (Haar) and Sym8 wavelets are members of the Daubechies family, designed for discrete wavelet transforms with compact support and varying smoothness.

1.3 What is a Wavelet Transform?

The wavelet transform (WT) analyzes a signal by projecting it onto scaled and shifted versions of a wavelet, enabling the examination of signal features at multiple scales. Unlike traditional Fourier transforms (FT), which represent signals using infinite-length sine and cosine functions, wavelets are localized in both time and frequency.

In essence, WT provide a way to analyze signals that have changing frequency content over time. This makes them especially useful in fields such as signal processing, image compression, and time-frequency analysis. There are various types of wavelets, each with unique properties suited for different applications.

1.4 Why Use Wavelet Transforms?

Wavelet transforms offer several unique advantages:

- **Multiresolution analysis:** Analyze signals at different scales or resolutions simultaneously.

- **Time-frequency localization:** Capture both when and what frequency components occur.

- **Sparse representation:** Many real-world signals have a sparse representation in the wavelet domain, leading to efficient compression and denoising.

- **Non-stationary signal analysis:** Time-frequency localization Ideal for signals whose frequency content changes over time, such as biomedical signals, financial data, and environmental sensor streams.

1.5 Applications of Wavelet Transforms

Wavelet transforms are widely used in real-world applications, including:

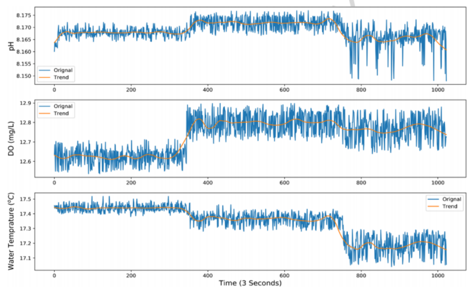
- **Image compression:** Used in JPEG2000 for better quality at higher compression ratios.

- **Denoising:** Removing noise from electrocardiogram (ECG), audio, or stock signals.
- **Trend analysis and forecasting:** Extracting long-term trends in time series for prediction and interpretation.
- **Feature extraction:** For machine learning application involving time series or images.
- **Fault detection:** Identifying abrupt discontinuities, anomalies, or changes in mechanical and electrical systems (e.g., via vibration signals).
- **Environmental monitoring:** Cleaning and analyzing data from distributed sensor networks.
- **Data compression:** Efficiently representing large datasets with minimal storage (e.g., signals, images, or video).
- **Data encryption and security:** Protecting data through wavelet-based transformations in secure transmission and storage.

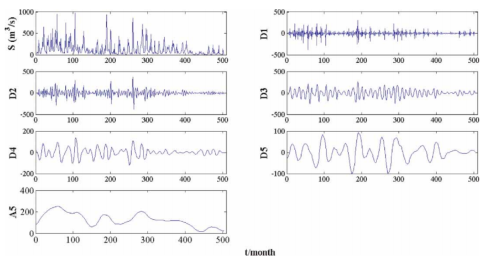
The following examples illustrate practical applications in real-world scenarios.

1. Noise removal and trend analysis

Two common applications of wavelets are denoising and trend analysis (Figure 1.2).



(a) Denoising

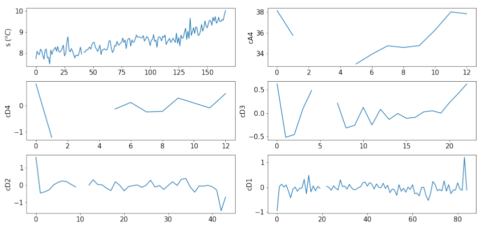


(b) Trend analysis

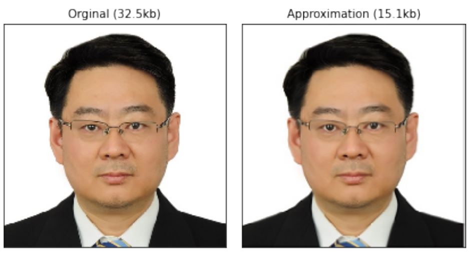
Figure 1.2: Wavelet transform application: (a) Denoising (Zhao et al. 2020), (b)Trend analysis and forecasting (Wei, Song, and Khan 2011).

2. Abrupt discontinuities, abnormal changes and image compression

Two common applications of wavelet transforms are outlier detection and image compression (Figure 1.3).



(a) Outlier detection



(b) Original approximation

Figure 1.3: Application of wavelet transforms: (a) detecting discontinuities and abnormal changes, and (b) performing image compression.

3. Wavelet-Based Image Steganography

Steganography is the practice of concealing information within other data, and wavelets provide an effective domain for this purpose. By embedding secret data in the wavelet coefficients of an image, we can achieve robust and imperceptible hiding. This technique leverages the multiresolution nature of wavelet transforms to distribute the hidden message across different frequency bands, making detection difficult while preserving image quality.

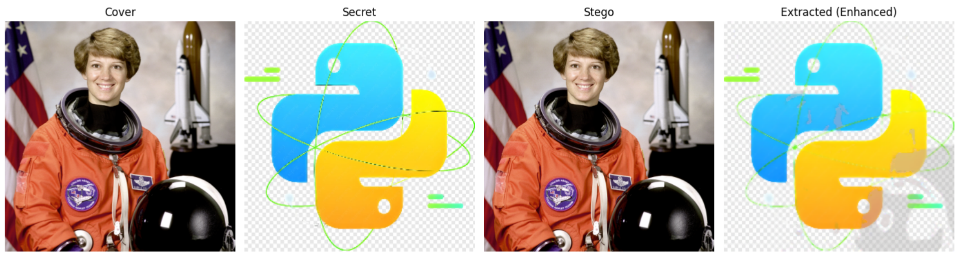


Figure 1.4: The Discrete Wavelet Transform (DWT) Based Steganography. The figure displays comprehensive visualization including: (1) original cover image, (2) secret image to be hidden, (3) stego-image with embedded secret showing PSNR values, and (4) extracted secret image.

Key steps in wavelet-based steganography include:

1. **Decomposition:** Apply a 2-D wavelet transform to the cover image, separating it into approximation and detail subbands.
2. **Embedding:** Modify selected wavelet coefficients (often in the detail subbands) to encode the secret message.
3. **Reconstruction:** Perform the inverse wavelet transform to generate the stego-image, which appears nearly identical to the original.
4. **Extraction:** The recipient applies the same wavelet transform to retrieve and decode the hidden data.

Benefits:

- **Robustness:** The hidden message can survive common image manipulations such as compression or noise addition.

- **Imperceptibility:** Changes are distributed across multiple scales, minimizing visible artifacts.

- **Scalability:** Different levels of wavelet decomposition can be used to balance capacity and security.

1.6 Wavelet Transform vs. Fourier Transform

While both the Fourier transform and wavelet transform are powerful tools for signal analysis, they differ fundamentally in how they represent signals.

1.6.1 Fourier Transform (FT)

The Fourier transform decomposes a signal into sine and cosine waves of different frequencies. It provides excellent frequency resolution but lacks time resolution—we know *what* frequencies are present but not *when* they occur. This makes FT ideal for analyzing stationary signals (signals whose frequency content does not change over time).

Key Properties:

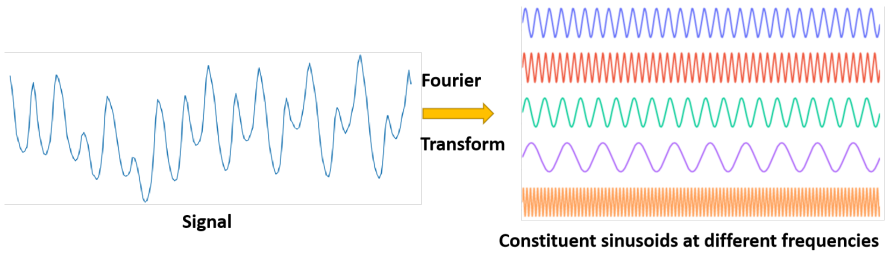


Figure 1.5: Fundamental concepts of the Fourier transform (FT), showing how a time-domain signal can be decomposed into a sum of sinusoidal components, thereby providing its frequency-domain representation.

- **Global Analysis:** Represents the entire signal as a sum of sinusoidal components.
- **Frequency Domain:** Provides precise frequency information.
- **Limitation:** Poor time localization—cannot identify when specific frequency events occur.

1.6.2 Short-Time Fourier Transform (STFT)

The Short-Time Fourier Transform is an extension of the Fourier transform that analyzes localized sections of a signal over time using a sliding window, offering limited time-frequency resolution.

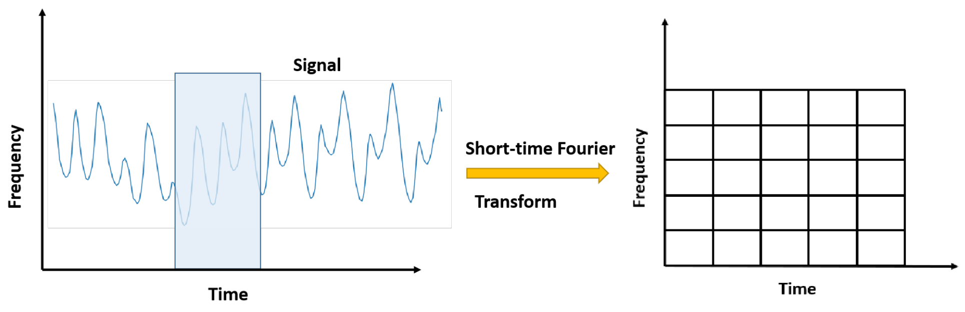


Figure 1.6: Illustration of the Short-Time Fourier Transform (STFT), highlighting time-localization by dividing the signal into short overlapping windows and analyzing each segment independently.

- STFT addresses the time-localization issue by dividing the signal into short segments (windows) and analyzing each segment separately.
- It provides some information about when and where different frequency components occur.
- However, the fundamental problem lies in choosing the window size:
- A narrow window provides good time resolution but poor frequency resolution.
- A wide window gives good frequency resolution but poor time resolution.
- As a result, STFT suffers from poor localization in both time and frequency.

1.6.3 Wavelet Transform (WT)

The wavelet transform, by contrast, decomposes a signal using localized waveforms (wavelets) that can be scaled and shifted. This provides both time and frequency information, allowing us to see *what* frequencies are present and *when* they occur. WT is particularly effective for non-stationary signals where frequency content varies with time.

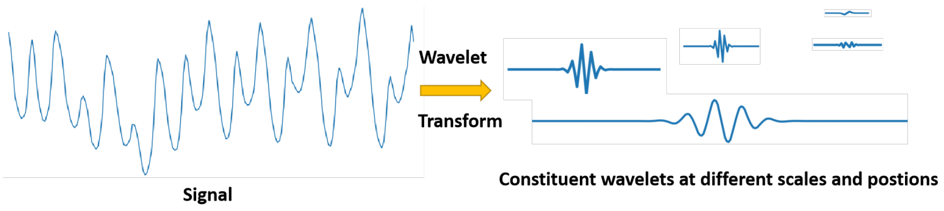


Figure 1.7: Illustration of the wavelet transform (WT), showing how a signal is decomposed into components at multiple scales/index{wavelet theory!multiple scales} to capture both time and frequency information, with Scale (or Dilation) parameters a controlling the wavelet width and Shift (or Translation) parameter b determining its time position.

The **wavelet transform**, also known as **Wavelet Analysis**, offers a modern alternative to the traditional Fourier Transform, especially for analyzing **non-stationary signals**.

At its core, wavelet analysis relies on two fundamental building blocks: the **Mother Wavelet** and the **Father Wavelet**.

- The **Mother Wavelet (Wavelet function)** captures the detail coefficients or high-frequency components of a signal of a signal.
- The **Father Wavelet (Scaling function)** represents the approximation coefficients or low-frequency components.

When analyzing a signal, the wavelet is modified by two parameters:

Two Basic Factors:

- **Scale (or Dilation) parameter** a**:** Controls how the wavelet is **stretched** or **compressed**. The relationship between **scale** a and **frequency** ω is inverse.
  - **Small scale** a Compressed wavelet   Rapidly changing details   **High frequency** ω
  - **Large scale** a Stretched wavelet   Slowly changing, coarse features   **Low frequency** ω
- **Shift (or Location) parameter** b**:** Controls where the wavelet is positioned in time (or space) while analyzing the signal.

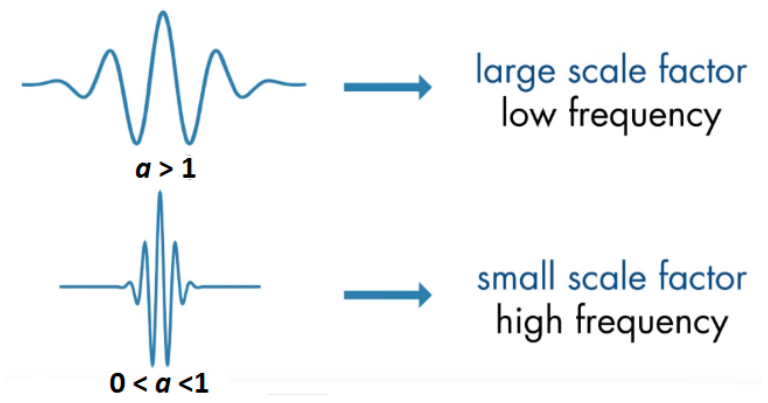


Figure 1.8: Illustration of the inverse relationship between scale (a) and frequency (ω), showing that larger scales correspond to lower frequencies and smaller scales correspond to higher frequencies.

The Wavelet Transform uses **inner products** to measure the similarity between the signal and scaled/shifted versions of a wavelet. This results in a set of **wavelet coefficients** that represent localized time-frequency information.

Key Properties:

- **Local Analysis:** Uses wavelets that are localized in both time and frequency.
- **Multi-resolution:** Analyzes signals at different scales, providing both coarse and fine detail.
- **Adaptability:** Suitable for transient signals, edges, and abrupt changes.

1.6.4 Comparison Summary

The Table 1.1 summarizes the key differences between the Fourier transform and the wavelet transform, highlighting their respective strengths and limitations in various applications.

Table 1.1: Comparison of Wavelet Transform and Fourier Transform.

Aspect	**Fourier Transform (FT)**	**Wavelet Transform (WT)**
Basis Functions	Sine and cosine waves	Localized wavelets
Time Localization	None (global)	Excellent (local)
Frequency Resolution	High	Variable (depends on scale)
Best For	Stationary signals	Non-stationary signals
Applications	Spectral analysis, filtering	Denoising, compression, edge detection
Computational Cost	FFT: O(N log N)	FWT{FWT}: O(N)

- **Fourier Transform:** Analyzing the frequency spectrum of a pure tone, studying periodic signals.

- **Wavelet Transform:** Detecting transients in seismic data, analyzing ECG signals, compressing images.

In summary, while the Fourier transform excels in frequency analysis of stationary signals, the wavelet transform provides superior performance for non-stationary signals requiring joint time-frequency representation.

1.6.5 Real-World Examples

The following example illustrates a comparison between analysis visualizations obtained through the Fourier transform and the wavelet transform.

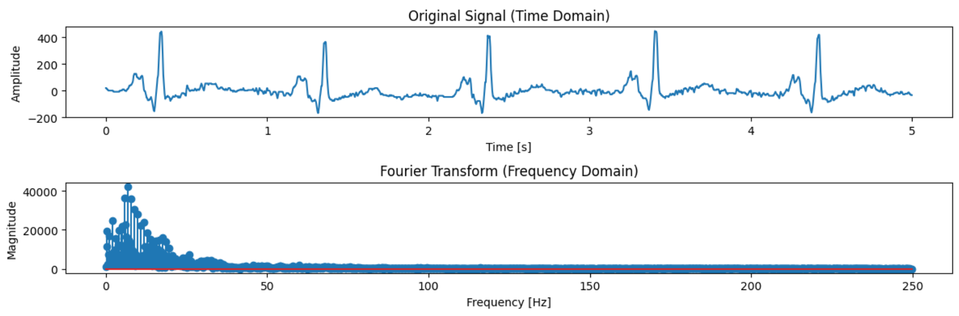


Figure 1.9: Time-domain ECG signal (top) and its corresponding frequency-domain representation obtained via Fourier Transform (bottom), showing the dominant frequency components concentrated in the low-frequency range below 50 Hz.

While both the Fourier transform (FT) and the wavelet transform (WT) are powerful signal processing techniques for ECG analysis, they offer complementary advantages. The Fourier transform (Figure 1.9) decomposes the signal into its constituent frequency components, revealing that ECG signals contain predominantly low-frequency information below 50 Hz, but it provides no temporal information about when these frequencies occur during the cardiac cycle. In contrast, the wavelet transform (Figure 1.10) provides a multiresolution time-frequency representation, simultaneously preserving both temporal localization and frequency information. This allows the wavelet approach to capture transient features like QRS complexes and identify precisely when specific frequency components appear in the signal. The approximation coefficients retain the overall ECG morphology and

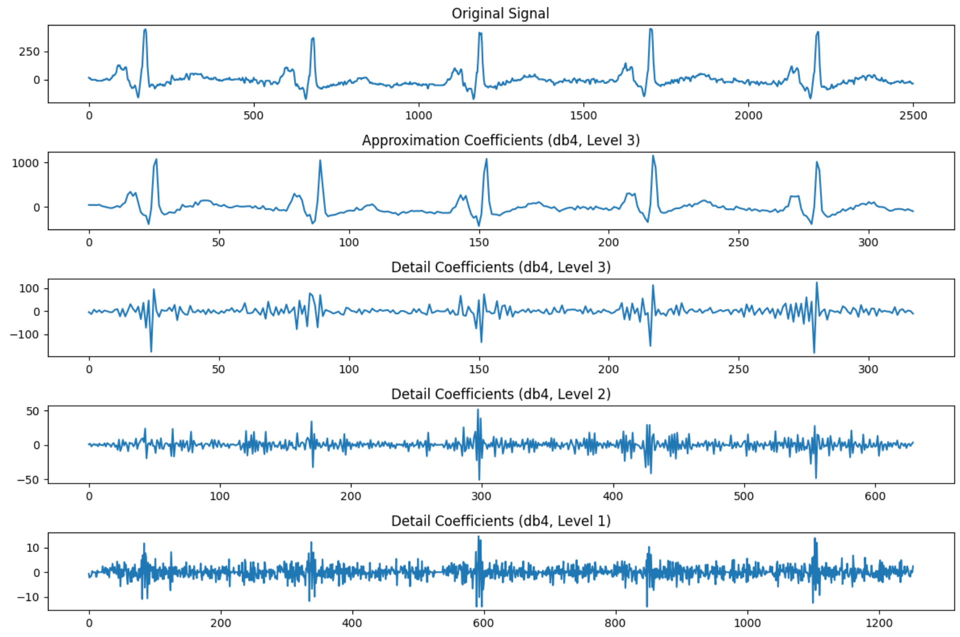


Figure 1.10: Wavelet decomposition of an ECG signal using the Daubechies 4 (db4) wavelet at three decomposition levels. The top panel shows the original signal, followed by the approximation coefficients (smooth, low-frequency components) at level 3, and detail coefficients at levels 3, 2, and 1 (capturing progressively higher-frequency features from coarse to fine resolution).

low-frequency trends, while the detail coefficients at different levels isolate high-frequency noise and artifacts at specific time points. This temporal localization makes wavelets particularly valuable for detecting arrhythmias, sudden signal changes, and non-stationary features that would be obscured in the purely frequency-domain representation of the Fourier transform.

1.7 Classifications of Wavelet Transforms

Wavelet transforms can be classified in several ways, each suited to different types of data and applications. The main classifications are:

1.7.1 1. Based on Continuity

1. Continuous Wavelet Transform (CWT)

The CWT analyzes a signal by computing wavelet coefficients at every possible scale and position. It provides a highly redundant but detailed representation, making it ideal for time-frequency analysis and feature detection.

Properties:

- Uses continuous scales and translations.
- Provides a smooth, continuous time-frequency map.
- Highly redundant—many coefficients represent overlapping information.

Applications:

- Transient detection in signals.
- Pattern recognition.
- Detailed time-frequency visualization.

2. Discrete Wavelet Transform (DWT)

The DWT uses a discrete set of scales and positions (typically powers of 2), resulting in a non-redundant, efficient representation of the signal. It is the foundation of most practical wavelet applications, including compression and denoising.

Properties:

- Uses dyadic scales (2^j) and integer translations.
- Decomposes signals into approximation and detail coefficients.
- Efficient and computationally fast.

Applications:

- Image and signal compression.
- Denoising and feature extraction.
- Data analysis and statistical modeling.

1.7.2 2. Based on Dimensionality

Wavelet transforms can also be categorized by the number of dimensions of the data being analyzed:

1. 1-D Wavelet Transform

The one-dimensional (1-D) wavelet transform is applied to one-dimensional signals such as time series, audio signals, or stock price data. It decomposes the signal into approximation and detail components at multiple scales, enabling multi-scale analysis of temporal patterns.

Applications:

- Time series forecasting.

- Audio compression and denoising.

- Biomedical signal processing (e.g., ECG, EEG).

2. 2-D Wavelet Transform

The 2-D wavelet transform is primarily used for image processing. It extends the 1-D transform by applying wavelet decomposition first along rows and then along columns of an image matrix. This results in four subbands: approximation, horizontal detail, vertical detail, and diagonal detail. It is widely used in image compression (e.g., JPEG2000), edge detection, and texture analysis.

3. N-D Wavelet Transform ($n > 2$)

The n-dimensional or multi-dimensional wavelet transform is used to analyze data in three or more dimensions, such as 3-D medical imaging (e.g., MRI or CT scans), video sequences, or volumetric data from scientific simulations. It generalizes the wavelet decomposition to higher-dimensional arrays by applying the transform along each dimension. This allows for efficient analysis and compression of complex multi-dimensional data structures.

1.7.3 Other Classifications of Wavelet Transforms

1. Based on Decomposition Levels

Wavelet transforms can also be categorized by how many levels or stages of decomposition are performed:

- **Single-Level (One-Stage) Decomposition**: A signal is decomposed only once into approximation and detail components. This is useful for simple applications where a coarse analysis is sufficient.
- **Multi-Level Decomposition**: The approximation component from each level is further decomposed in a recursive manner. This allows for a hierarchical, multi-scale representation of the signal or image and is fundamental to multiresolution analysis. **2. Based on the Type of Discrete Wavelet Transform (DWT)**

There are several variations of the Discrete Wavelet Transform (DWT), each offering different characteristics and benefits:

- **Discrete Wavelet Transform (DWT)**: The standard and most widely used form of the wavelet transform for both 1-D and 2-D signals. It performs subsampling and provides a compact, non-redundant representation.
- **Dual-Tree Complex Wavelet Transform (DT-CWT)**: An enhanced version of DWT that uses two parallel filter banks to provide approximate shift invariance and improved directional selectivity, especially useful in image and video processing.
- **Stationary Wavelet Transform (SWT)**: Also known as the undecimated wavelet transform, it avoids downsampling, resulting in a redundant representation that is translation-invariant—important for denoising and edge detection.
- **Multiresolution Analysis (MRA)**: A theoretical framework underpinning many wavelet-based techniques. It decomposes signals across multiple scales using nested function spaces, often implemented using DWT or MODWT.
- **Wavelet Packet Transform (WPT)**: An extension of DWT in which both approximation and detail coefficients are recursively decomposed. It offers a richer analysis and is useful for feature extraction and classification tasks.
- **Maximum Overlap Discrete Wavelet Transform (MODWT)**: A variant of SWT that corrects for phase shifts and allows for better alignment of wavelet coefficients with the original signal. It is particularly effective in time series analysis.
- **Multiresolution Analysis based on MODWT (MODWTMRA)**: Applies the MODWT in a multilevel framework to provide a multiresolution decomposition of a signal, commonly used in statistical signal analysis and trend decomposition.

1.8 Summary

This introductory chapter addressed several fundamental questions to build a foundational understanding of wavelets and their transformations:

- **What are Wavelets?** Small, localized wave-like functions used to represent data at multiple scales and resolutions.
- **Why do we use Wavelets?** Because they offer both time and frequency localization, making them ideal for analyzing non-stationary signals, detecting abrupt changes, compressing data, and denoising.
- **How do we use Wavelets?** Through the wavelet transform (WT), which decompose signals into approximation and detail components using scaling and wavelet functions.
- **How does the wavelet transform (WT) differ from the Fourier transform (FT)?** Unlike FT, which only provides frequency information, WT provides both time and frequency resolution through multiresolution analysis.
- **How are wavelet transforms (WT) classified?** Wavelet transforms can be classified based on continuity (CWT vs. DWT), dimensionality (1-D, 2-D, n-D), decomposition levels (single vs. multi-level), and transform types (e.g., DWT, SWT, DT-CWT, MODWT).

1.9 Exercises and Quizzes

1.9.1 Quick Quizzes

1. **What is the main difference between the Fourier transform (FT) and the Wavelet Transform (WT)?**

   A. Wavelet transform only works on periodic signals

   B. Fourier transform has variable time resolution

   C. Wavelet transform provides time-frequency localization

   D. Fourier transform uses wavelets for basis functions

2. **What is a “Mother Wavelet”?**

   A. A high-pass filter

   B. The original wavelet used to generate other wavelets

   C. A type of noise-reducing signal

   D. A component of the Fourier basis

1.9.2 Exercises

1. Load a time series (e.g., daily temperatures or stock prices).
2. Plot the original time series.
3. Discuss how wavelet analysis might help you understand the local behavior of the series.

Answers:

1. C
2. B

References

Antoine, Jean-Pierre, Romain Murenzi, Pierre Vandergheynst, and Syed Twareque Ali. 2004. "Two-Dimensional Wavelets and Their Relatives." *Two-Dimensional Wavelets and Their Relatives*, September. https://doi.org/10.1017/CBO9780511543395.

Barneva, Reneta P., Valentin E. Brimkov, and Josef Šlapal, eds. 2014. *Combinatorial Image Analysis*. Vol. 8466. Springer International Publishing. https://doi.org/10.1007/978-3-319-07148-0.

Beylkin, G., R. Coifman, and V. Rokhlin. 1991. "Fast Wavelet Transforms and Numerical Algorithms i." *Communications on Pure and Applied Mathematics* 44 (March): 141–83. https://doi.org/10.1002/CPA.3160440202;REQUESTEDJOURNAL:JOURNAL:10970312;WGROUP:STRING:PUBLICATION.

Coifman, Ronald R., and Mladen Victor Wickerhauser. 1992. "Entropy-Based Algorithms for Best Basis Selection." *IEEE Transactions on Information Theory* 38: 713–18. https://doi.org/10.1109/18.119732.

Daubechies, Ingrid. 1988. "Orthonormal Bases of Compactly Supported Wavelets." *Communications on Pure and Applied Mathematics* 41 (October): 909–96. https://doi.org/10.1002/CPA.3160410705.

———. 1992. *Ten Lectures on Wavelets*. Society for Industrial. https://doi.org/10.1137/1.9781611970104.

DeVore, R. A., S. V. Konyagin, and V. N. Temlyakov. 1998. "Hyperbolic Wavelet Approximation." *Constructive Approximation* 14 (January): 1–26. https://doi.org/10.1007/S003659900060/METRICS.

Donoho, David L., and Jain M. Johnstone. 1994. "Ideal Spatial Adaptation by Wavelet Shrinkage." *Biometrika* 81 (September): 425–55. https://doi.org/10.1093/BIOMET/81.3.425.

Forster, Brigitte. 2014. "Splines and Multiresolution Analysis." In *Handbook of Mathematical Methods in Imaging*, 1–38. Springer Berlin Heidelberg.

https://doi.org/10.1007/978-3-642-27795-5_28-5.

Gonzales, R. E, and R. C Woods. 2018. "Digital Image Processing Digital Image Fundamental." *Radiologic Technology* 4th editio: 1024.

Lee, Gregory, Ralf Gommers, Filip Waselewski, Kai Wohlfahrt, and Aaron O'Leary. 2019. "PyWavelets: A Python Package for Wavelet Analysis." *Journal of Open Source Software* 4 (36): 1237. https://doi.org/10.21105/joss.01237.

Mallat, S. 1989. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (7): 674–93. https://doi.org/10.1109/34.192463.

Mallat, S. G.. 1999. "A Wavelet Tour of Signal Processing," 637.

Mallat, Stéphane. 1998. "A Wavelet Tour of Signal Processing (1st Edition)." Academic Press. https://www.abebooks.com/first-edition/Wavelet-Tour-Signal-Processing-Mallat-Stephane/30519382007/bd.

———. 2009. *A Wavelet Tour of Signal Processing (3rd Edition)*. Elsevier. https://doi.org/10.1016/B978-0-12-374370-1.X0001-8.

Meyer, Yves. 1993. *Wavelets and Operators*. Edited by D. H. Salinger. Cambridge University Press. https://doi.org/10.1017/cbo9780511623820.

Nason, G. P., and B. W. Silverman. 1995. "The Stationary Wavelet Transform and Some Statistical Applications." In *Wavelets and Statistics*, 281–99. Springer New York. https://doi.org/10.1007/978-1-4612-2544-7_17.

Perona, Pietro, and Jitendra Malik. 1990. "Scale-Space and Edge Detection Using Anisotropic Diffusion." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12: 629–39. https://doi.org/10.1109/34.56205.

Pesquet, J.-C., H. Krim, and H. Carfantan. 1996. "Time-Invariant Orthonormal Wavelet Representations." *IEEE Transactions on Signal Processing* 44 (8): 1964–70. https://doi.org/10.1109/78.533717.

Ravier, Philippe, and Pierre-Olivier Amblard. 2001. "Wavelet Packets and de-Noising Based on Higher-Order-Statistics for Transient Detection." *Signal Processing* 81 (9): 1909–26. https://doi.org/https://doi.org/10.1016/S0165-1684(01)00088-3.

Rioul, O., and M. Vetterli. 1991. "Wavelets and Signal Processing." *IEEE Signal Processing Magazine* 8 (4): 14–38. https://doi.org/10.1109/79.91217.

Rosiene, Carolyn Pe, and Truong Q. Nguyen. 1999. "Tensor-Product Wavelet

Vs. Mallat Decomposition: A Comparative Analysis." *Proceedings - IEEE International Symposium on Circuits and Systems* 3. https://doi.org/10.1109/ISCAS.1999.778877.

Starck, Jean Luc, Fionn Murtagh, and Jalal M. Fadili. 2010. "Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity." *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity* 9780521119139 (January): 1–328. https://doi.org/10.1017/CBO9780511730344.

Taubman, David S., and Michael W. Marcellin. 2002. "JPEG2000 Image Compression Fundamentals, Standards and Practice." *JPEG2000 Image Compression Fundamentals, Standards and Practice*. https://doi.org/10.1007/978-1-4615-0799-4.

Unser, Michael, and Thierry Blu. 2003. "Wavelet Theory Demystified." *IEEE Transactions on Signal Processing* 51: 470–83. https://doi.org/10.1109/TSP.2002.807000.

Velisavljević, Vladan, Baltasar Beferull-Lozano, Martin Vetterli, and Pier Luigi Dragotti. 2006. "Directionlets: Anisotropic Multidirectional Representation with Separable Filtering." *IEEE Transactions on Image Processing* 15 (July): 1916–33. https://doi.org/10.1109/TIP.2006.877076.

Vetterli, Martin, and Jelena Kovačević. 1995. "Wavelets and Subband Coding." *Book*, 1–519.

Wei, Shouke, Jinxi Song, and Nasreen Islam Khan. 2011. "Simulating and Predicting River Discharge Time Series Using a Waveletneural Network Hybrid Modelling Approach." *Hydrological Processes* 26 (2): 281–96. https://doi.org/10.1002/hyp.8227.

Westerink, P. H. 1989. "Subband Coding of Images." Ph.D. dissertation, Delft University of Technology, Department of Electrical Engineering, Information Theory Group.

Wickerhauser, Mladen Victor. 1996. "Adapted Wavelet Analysis : From Theory to Software." *Adapted Wavelet Analysis*, April. https://doi.org/10.1201/9781439863619.

Yu, Jinyong, Yiming Sun, and Zhengchao Li. 2018. "Event-Based Robust Filter Design for a Class of State-Dependent Uncertain Systems with Network Transmission Delay Under a Unified Framework." *Signal Processing* 148 (July): 56–67. https://doi.org/10.1016/J.SIGPRO.2018.01.023.

Zhao, Jindong, Shouke Wei, Xuebin Wen, and Xiuqin Qiu. 2020. "Analysis and Prediction of Big Stream Data in Real-Time Water Quality Monitoring System." *Journal of Ambient Intelligence and Smart Environments* 12 (5): 393–406. https://doi.org/10.3233/ais-200571.

Index

VOLUME I

Wavelet Transforms with Python bridges theory and practice in modern signal processing, image analysis, and time-series modeling. This volume provides:

- Mathematical foundations of wavelet theory
- Scaling functions and orthonormal wavelet bases
- Multiresolution Analysis (MRA) framework
- 1D, 2D, and nD Discrete Wavelet Transform (DWT)
- Stationary Wavelet Transform (SWT)
- Wavelet Packet Transform (WPT)
- Visualization and interpretation of wavelet coefficients.

About the Author

Shouke Wei earned his Ph.D. from Brandenburg University of Technology Cottbus–Senftenberg (Germany). and conducted postdoctoral research at the Swiss Federal Institute of Aquatic Science and Technology (Eawag, Switzerland). He has held research postions at the University of British Columbia (Canada) and distinguished and adjunct professorships at multiple institutions (China).

Dr. Wei's research focuses on rigorous, reproducible, and practically deployable wavelet-based signal processing methods, with applications in environmental monitoring, robotics, and predictive analytics.

https://press.deepsim.ca

9 781069 928412