

Spectral Analysis of Time-series

Anshul Thakur Andrew Creagh Prof. David A. Clifton

Department of Engineering Science
University of Oxford

Centre for Doctoral Training in Healthcare Innovation

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

Short-term Fourier Transform

Further Reading

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

Short-term Fourier Transform

Further Reading

What are Transforms?



- ▶ Transform data from one domain to other domain
 - ▶ E.g. Time domain to frequency domain
- ▶ Why transform data?
 - ▶ Analysis of data may be easier in transformed domain
 - ▶ Certain operations may only be feasible in transformed domain
- ▶ Linear Transformation functions are invertible
- ▶ Deep neural networks are non-linear non-invertible transforms!!!

Transforms around us!



- ▶ **Discrete Cosine Transform:** Represents a finite signal as a combination of cosine functions of different frequencies
- ▶ Allows compression of signals. Used in:
 - ▶ digital images ([JPEG](#))
 - ▶ digital video ([MPEG](#))
 - ▶ digital audio ([MP3](#), Dolby Digital)
 - ▶ speech coding
- ▶ Made streaming music, games and Netflix possible

Combination of Vectors



- ▶ Let $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \dots \mathbf{v}_n\}$ be a set of N vectors and $\{\alpha_1, \alpha_2, \alpha_3 \dots \alpha_n\}$ be a set of N scalars

- ▶ **Linear combination:** $\mathbf{y} = \sum_{i=1}^N \alpha_i \mathbf{v}_i$

- ▶ **Affine combination:** $\mathbf{y} = \sum_{i=1}^N \alpha_i \mathbf{v}_i$ where $\sum_{i=1}^N \alpha_i = 1$

- ▶ **Conic combination:** $\mathbf{y} = \sum_{i=1}^N \alpha_i \mathbf{v}_i$ where $\alpha_i \geq 0$

- ▶ **Convex combination:** $\mathbf{y} = \sum_{i=1}^N \alpha_i \mathbf{v}_i$ where $\sum_{i=1}^N \alpha_i = 1$ and $\alpha_i \geq 0$

Transform your data



- Let $\mathbf{x} \in \mathbb{R}^d$ be an input vector and $\mathbf{W} \in \mathbb{R}^{d \times m}$ be a transformation matrix, a simple linear transform is:

$$\mathbf{y} = \mathbf{W}^T \mathbf{x} \quad (1)$$

- Random Projections:** Project data to low dimensional space using Random Gaussian matrix

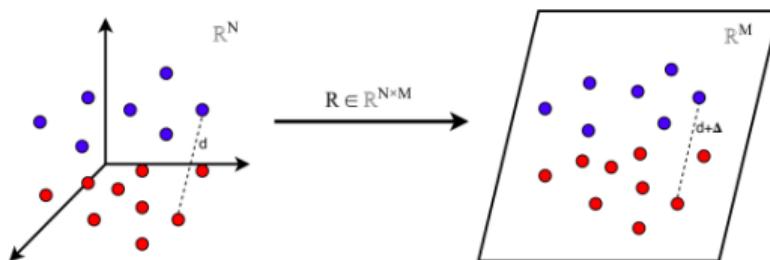


Figure: Random projections for dimensional reduction.

Transform Your Data



- ▶ Represent \mathbf{x} as linear combination of rows of \mathbf{W} :

$$\mathbf{x} = \sum_{i=1}^m y_i \mathbf{w}_i \tag{2}$$

Now, we can represent input vector \mathbf{x} as coefficients $[y_1, y_2, y_3, \dots, y_m]$

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

Short-term Fourier Transform

Further Reading

Types of Transforms



- ▶ Transforms can be categorised based on the **nature of transformation matrix**
 - ▶ **Analytical transform:** \mathbf{W} is **analytically** determined
 - ▶ **Learned transform:** \mathbf{W} is **learned** from the **data** itself
- ▶ Analytical transforms:
 - ▶ Data **independent** and universal
 - ▶ E.g. **Fourier** and **cosine** transforms where signal is decomposed onto sinusoidal waves of different frequencies
- ▶ Learned transforms:
 - ▶ Data **dependent**
 - ▶ \mathbf{W} contains basis of a subspace or exemplars
 - ▶ E.g. **PCA** as **principle components** are learned from the data
 - ▶ **Dictionary learning:** Learn transformation matrices or dictionaries from data and decompose data onto the atoms of these dictionaries

Types of Transforms: Deep Learning



- ▶ Deep neural networks can be seen as **cascades** of learned transforms:

$$\mathbf{y} = \mathbf{W}_3 \mathbf{W}_2 \mathbf{W}_1 \mathbf{x} \quad (3)$$

But this is linear! (Deep dictionary learning)

$$\mathbf{y} = \text{RELU}(\mathbf{W}_3 \text{RELU}(\mathbf{W}_2 \text{RELU}(\mathbf{W}_1 \mathbf{x}))) \quad (4)$$

- ▶ DNNs **learn** these transforms **simultaneously** using gradient descent
- ▶ Purpose is to learn embedding or **y**-space where classes are separable

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

Short-term Fourier Transform

Further Reading

Why Fourier Analysis?



- ▶ In healthcare, we often deal with 1d signals or time-series data:
 - ▶ EEG, ECG and SPO₂, Blood pressure
 - ▶ Speech signals
 - ▶ Breathing sound recordings
- ▶ Limited features can be extracted from signals in time-domain:
 - ▶ Zero crossing rate, autoregression coefficients and slope sign changes
 - ▶ How far can we really go with time-domain analysis?
 - ▶ Deep neural networks armed with Conv-1D filters can extract meaningful features
- ▶ Fourier analysis helps us to represent a signal as linear combination of sinusoidal components of different frequencies
 - ▶ Provides the frequency profile of a signal

Why Fourier Analysis?

- ▶ time-domain analysis shows how a signal changes over time
- ▶ frequency-domain analysis shows how the signal's energy is distributed over a range of frequencies
- ▶ frequency-domain representation also includes information on the phase shift
 - ▶ Phase tells you how all the frequency components align in time

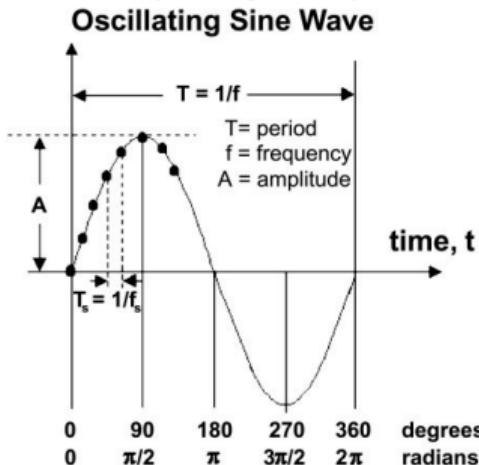


Figure: Phase and amplitude of a sine wave.

Fourier Series



- ▶ Any periodic signal can be represented as combination of sines and cosines:

$$f(t) = \frac{1}{2}A_0 + \sum_{n=1}^{\infty} A_n \cos(n\omega t) + \sum_{n=1}^{\infty} B_n \sin(n\omega t) \quad (5)$$

Fourier Series



- ▶ Coefficients can be worked out as:

$$A_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos(n\omega t) dt, \quad n = 0, 1, 2, 3 \dots \quad (6)$$

$$B_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(n\omega t) dt, \quad n = 1, 2, 3 \dots \quad (7)$$

- ▶ We are analysing $f(t)$ as mixtures of sines and cosines with **fundamental frequency** ω and **harmonics** $2\omega, 3\omega$ etc.
- ▶ $\omega = 2\pi/T$, where T is period of the signal
- ▶ $T = 1/f$, f is frequency of the signal and $\omega = 2\pi f$

Fourier Series: New Basis

- We have used sines and cosines of different frequencies as basis
- An alternation formulation based on different set of basis functions i.e. exponentials

$$e^{ix} = \cos x + i \sin x \quad (8)$$

- A representation based on this family of functions is called complex Fourier series:

$$f(x) = \sum_{n=-\infty}^{\infty} C_n e^{inx} \quad (9)$$

$$C_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-inx} dx \quad (10)$$

- C_n are normally complex

Fourier Transform



- ▶ Fourier transform and its associated inverse:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad (11)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} d\omega \quad (12)$$

- ▶ $F(\omega)$ is **frequency spectrum** of $f(t)$
 - ▶ No longer representing signal as sum of fundamental frequency and harmonics
- ▶ Can make analysis and certain operations easier:
 - ▶ Transform to frequency domain, remove higher frequency bands and reconstruct

Fourier Transform: Complex Frequency Spectrum



- ▶ Sinusoids are complex, so $F(\omega)$ will be **complex**
- ▶ Complex numbers: $z = x + iy$, x being **real** and y is **imaginary**
 - ▶ Magnitude of $z = \sqrt{x^2 + y^2}$, phase of $z = \tan^{-1}(y/x)$
- ▶ Similarly, we can compute magnitude spectrum and phase of $F(\omega)$
- ▶ Most of the spectral analysis is limited to magnitude spectrum
- ▶ But **phase** has **more information** about signal!

Importance of Phase



(A) Image A



(B) Image B



(C) Mag. of A and phase of B



(D) Mag. of B and phase of A



Figure: Impact of phase on reconstruction of an image.

Discarding Negative Frequencies

- ▶ $F(\omega)$ contains **infinite** frequencies and ω can be **negative**
 - ▶ $\cos(-\omega t) = \cos(\omega t)$ and $\sin(-\omega t) = -\sin(\omega t)$

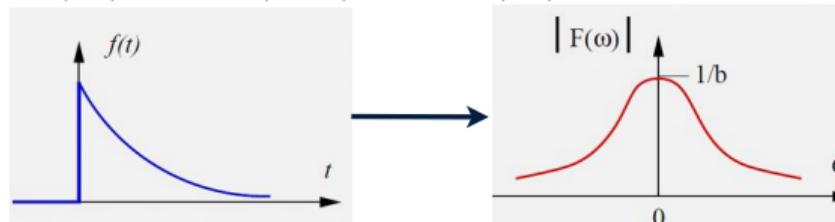


Figure: Frequency spectrum for positive frequencies is mirror of spectrum for negative frequencies.

- ▶ Just discard the negative frequency components from frequency spectrum

Frequency spectrum of EEG signal

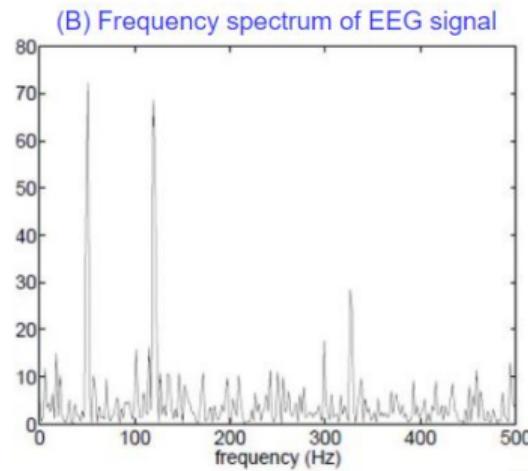
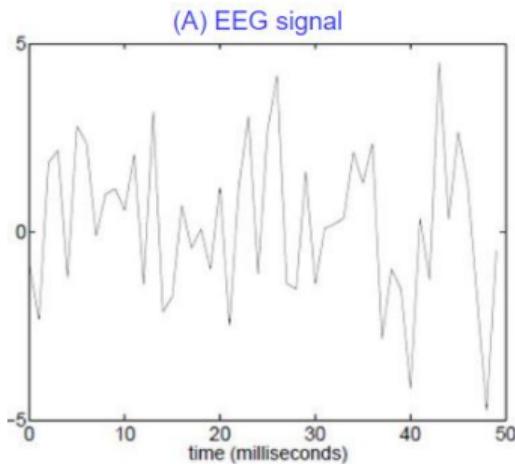


Figure: Frequency spectrum of EEG signal.

Table of Contents



Transforms

Types of Transforms

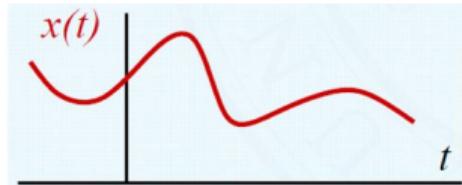
Fourier Transform

Discrete Fourier Transform

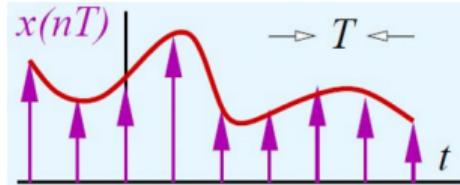
Short-term Fourier Transform

Further Reading

Discrete Signals



(A) Continuous Signal



(B) Discrete Signal

Figure: Discrete vs continuous signal.

- We work on **sampled** signals
 - **Sampling:** Record the values at fixed interval of time
 - **Sampling frequency (f_s):** Number of samples obtained in 1 second. E.g. 20 kHz implies 20,000 samples are obtained in 1 second

Nyquist-Shannon Sampling Theorem



- ▶ If $x(t)$ contains no frequency higher than B hertz, perfect reconstruction is guaranteed if $B < f_s/2$
- ▶ **Nyquist Frequency:** $f_s/2$, Highest frequency represented at sample rate f_s

Discrete Fourier Transform



- Discrete Fourier Transform and its inverse:

$$X[k] = \sum_{n=0}^{N-1} x[n] \exp\left(\frac{-j2\pi kn}{N}\right) \quad (13)$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \exp\left(\frac{j2\pi kn}{N}\right) \quad (14)$$

- Fast Fourier Transform (FFT) is an efficient algorithm for computing DFT
- **FFT:** Transforms a series of 2^n time-series data into frequency spectrum of 2^n components
- 2^n frequency components cover frequency range $[0, f_s]$

Discrete Fourier Transform



Practicalities of using FFT



- ▶ Suppose there is a signal sampled at 1000 Hz and it has 512 (2^9) samples
- ▶ 512 frequency components between $[0, fs]$
- ▶ Each component covers $1000/512 = 1.95$ Hz

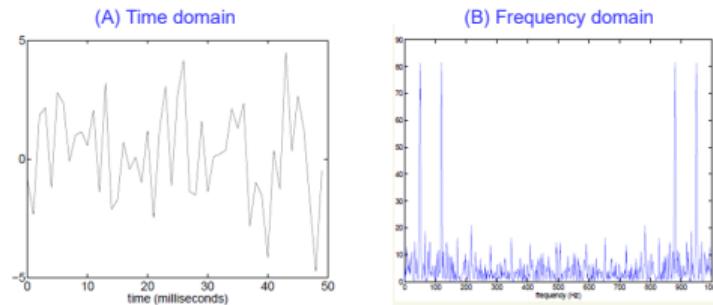


Figure: Time domain to frequency domain.

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

Short-term Fourier Transform

Further Reading

Short-term Fourier Transform

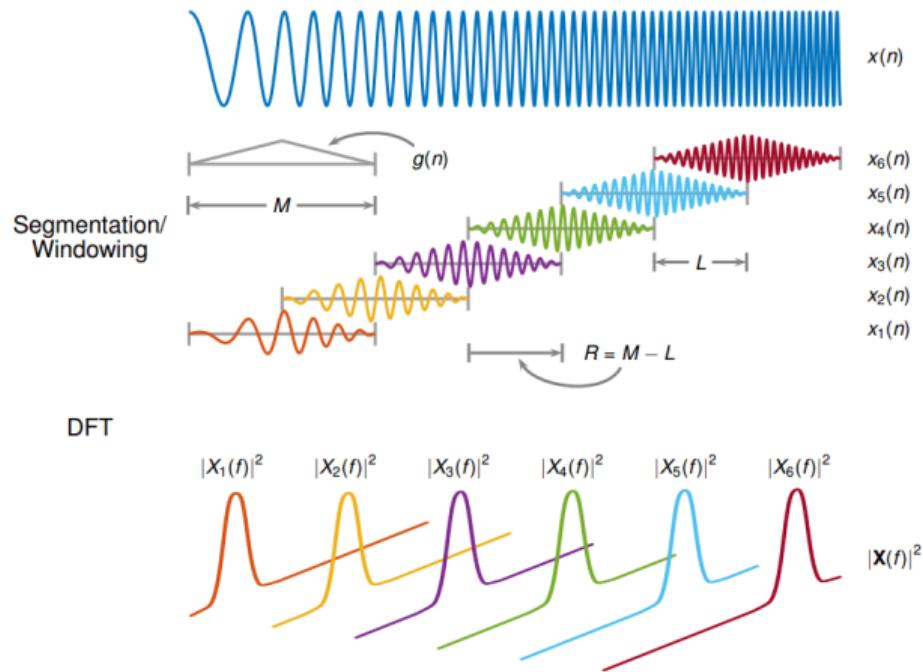


Figure: Short-term Fourier Transform

Short-term Fourier Transform

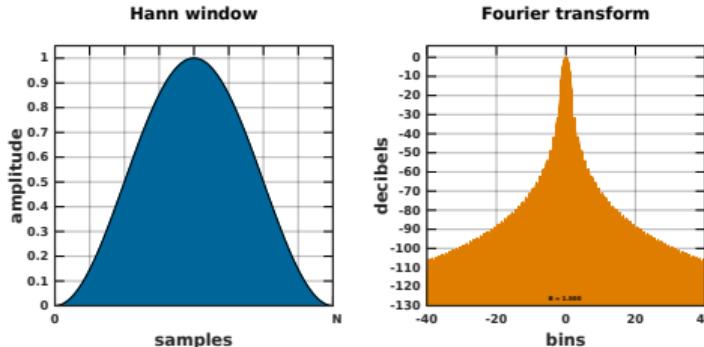


Figure: Windowing function: Hann window

- ▶ DFT can't capture **time-frequency modulations** in non-stationary signals
- ▶ **Short-term Fourier transform (STFT):**
 - ▶ Divide input signal into **overlapping segments** of M samples
 - ▶ **Analysis window** is applied to each segment to avoid **spectral ringing**
 - ▶ DFT is applied to each segment and **magnitude spectrum** is computed
 - ▶ $\mathbf{X}(f) = [\mathbf{x}_1(f) \ \mathbf{x}_2(f) \ \mathbf{x}_3(f) \dots \mathbf{x}_k(f)]$

Windowing

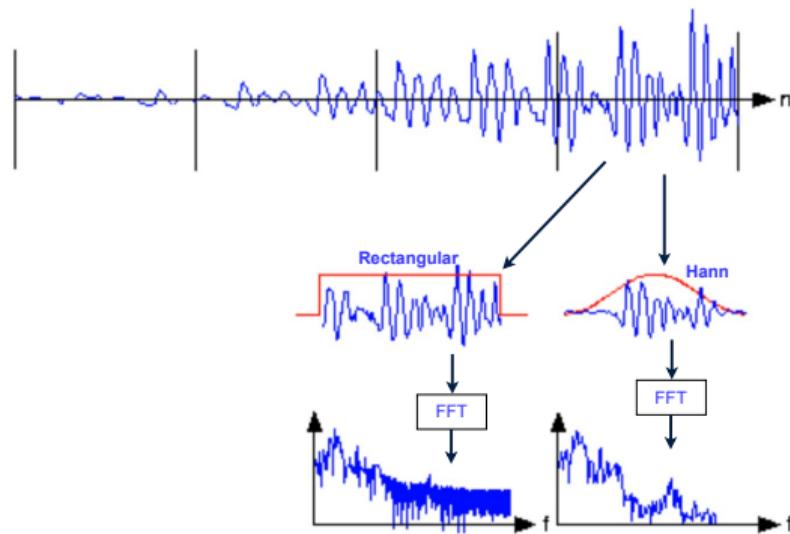


Figure: Difference in frequency profiles due to rectangular and Hann window.

- ▶ Signal may change **very abruptly** at the frame's edges
- ▶ Transform of such a segment reveals a **curious oscillation**
- ▶ Due to proper windowing, signal **decays gracefully** as it nears the edges

Spectrogram

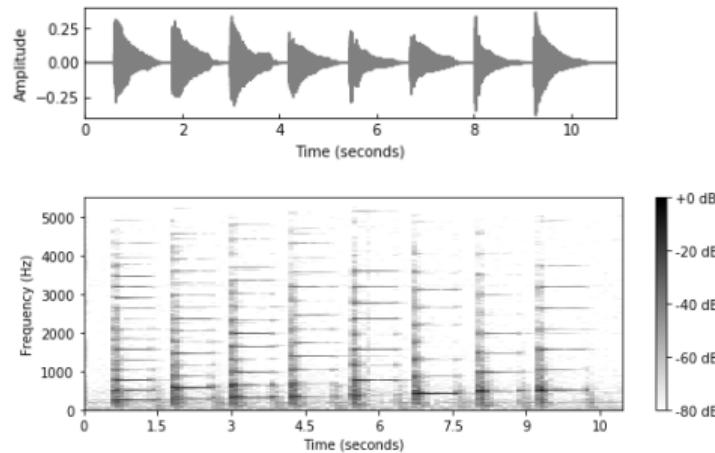


Figure: Spectrogram of an audio file containing C-major scale on piano.

Play

Why Spectrograms?



- ▶ Provides a way to analyse complex signals such as speech and soundscapes
- ▶ Spectrograms and features derived from it have provide state-of-the-art performance in many acoustic classification tasks
- ▶ Spectrograms allow us to use classic CNN architectures for 1d signal classification

Uncertainty in STFT: Time-frequency Resolution

- Smaller frame size or analysis window results in better time resolution but poor frequency resolution and vice-versa

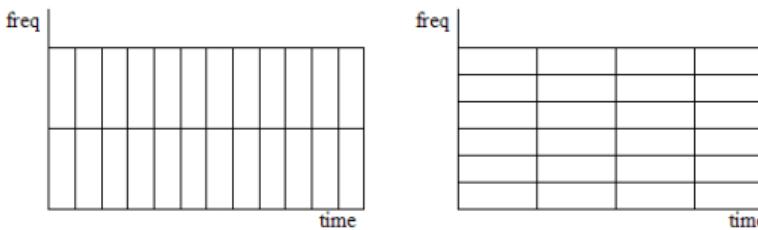


Figure: Conflict between time and frequency resolution.

- N samples in a window produces $N/2$ unique coefficients that represent frequencies between $[0, f_s/2]$
- Two coefficients are f_s/N Hz apart
- Increasing N will improve frequency resolution but decrease time-resolution

Why Phase is Ignored?

- **Phase Wrapping problem:** ARCTANGENT function can output values in $[-\pi, \pi]$. However, the magnitude of phase doesn't fit this range
 - So, ARCTANGENT wraps the phase in $[-\pi, \pi]$

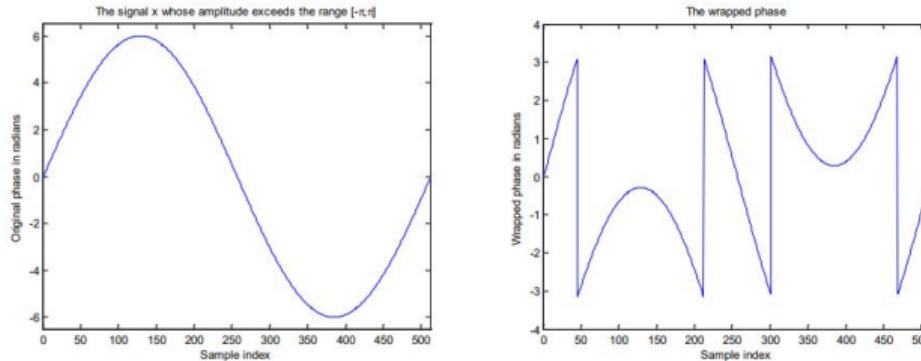


Figure: Phase wrapping problem.

- Difficult to recover phase
- Methods exists and phase has shown some amazing performance in speech and speaker recognition tasks

Why Phase is Ignored?

- ▶ Phase unwrapping methods such as all-pole group delay are **susceptible** to noise
- ▶ **Limiting** their use in many real-world scenarios
- ▶ So, should we just accept that phase has to be ignored?
- ▶ **Not really:** This loss of important phase information provides motivation for using **raw waveforms** as input to neural networks (Conv-1d networks)

Table of Contents



Transforms

Types of Transforms

Fourier Transform

Discrete Fourier Transform

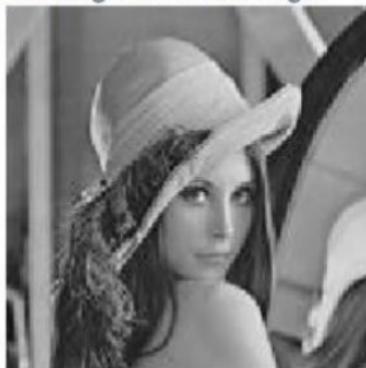
Short-term Fourier Transform

Further Reading

Further Reading: Discrete Cosine Transforms



- Original Lena image



- 2D DCT



Figure: DCT and Compression.

Further Reading: Multi-resolution Spectrograms

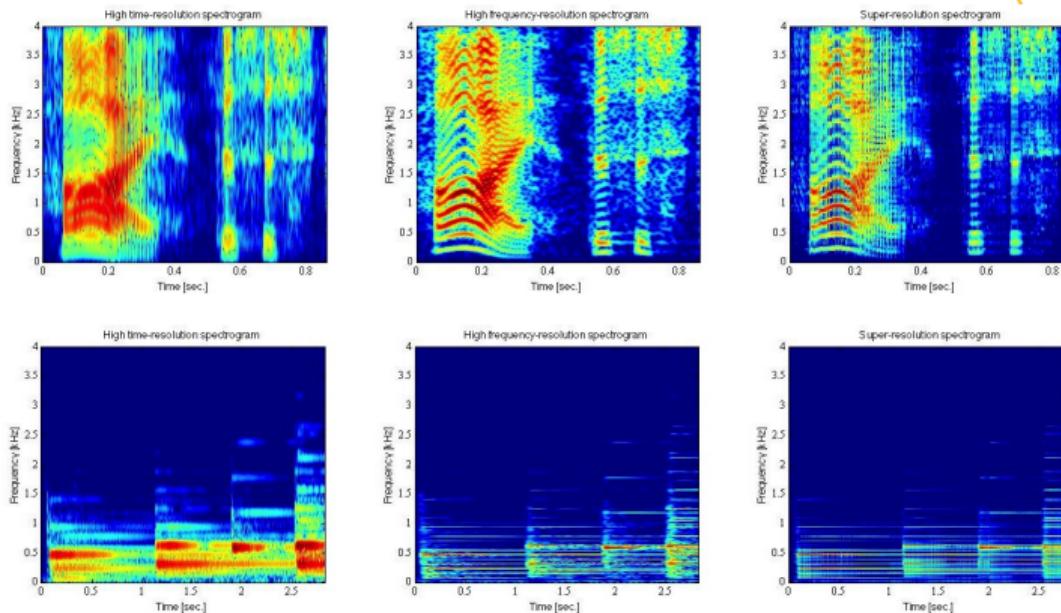


Figure: Multi-resolution spectrograms: Better time and frequency resolution.

Credit: SUPER-RESOLUTION SPECTROGRAM USING COUPLED PLCA, Nam et. al

Further Reading: Wavelet Transforms

- ▶ **Wavelet transform:** Good time resolution for high-frequency events and good frequency resolution for low-frequency events
- ▶ **Wavelet:** A rapidly decaying wave like oscillation with zero mean



Figure: Wavelet.

Wavelets

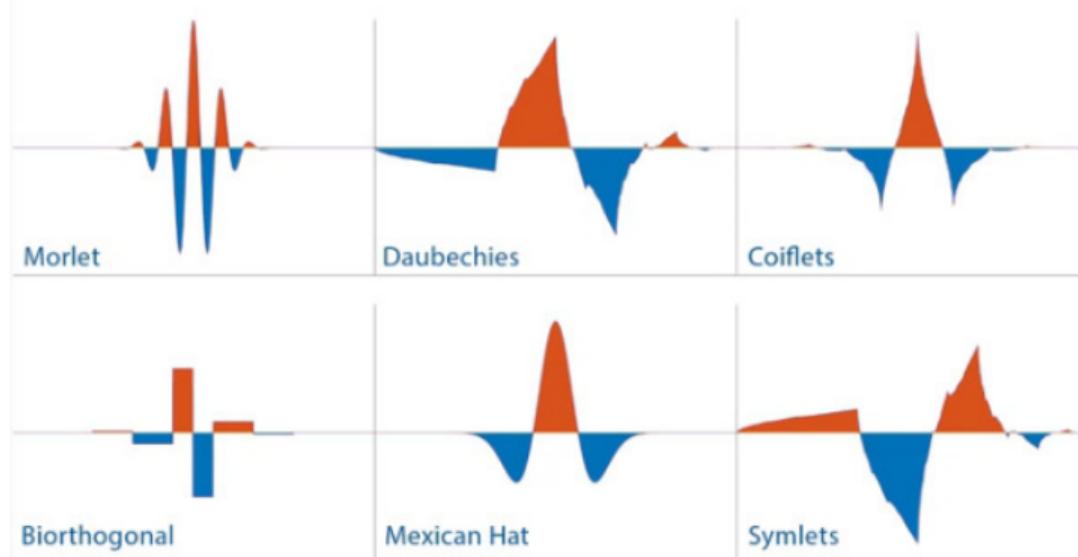


Figure: Some common wavelets.

Wavelets: Scaling and Shifting

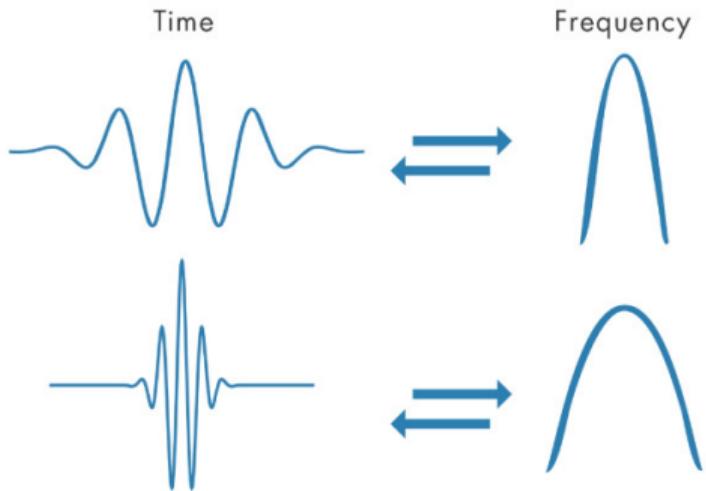


Figure: Scaling of wavelets.

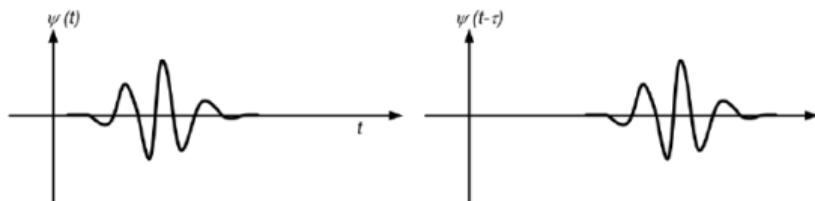


Figure: Shifting of wavelets.

Continuous Wavelet Transform

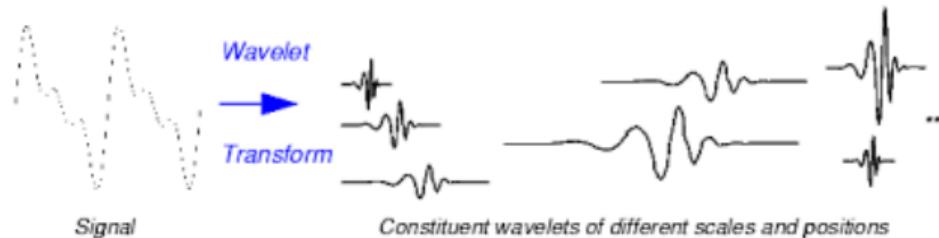


Figure: Representing a signal as combination of scaled and shifted wavelets.

STFT vs Continuous Wavelet Transform

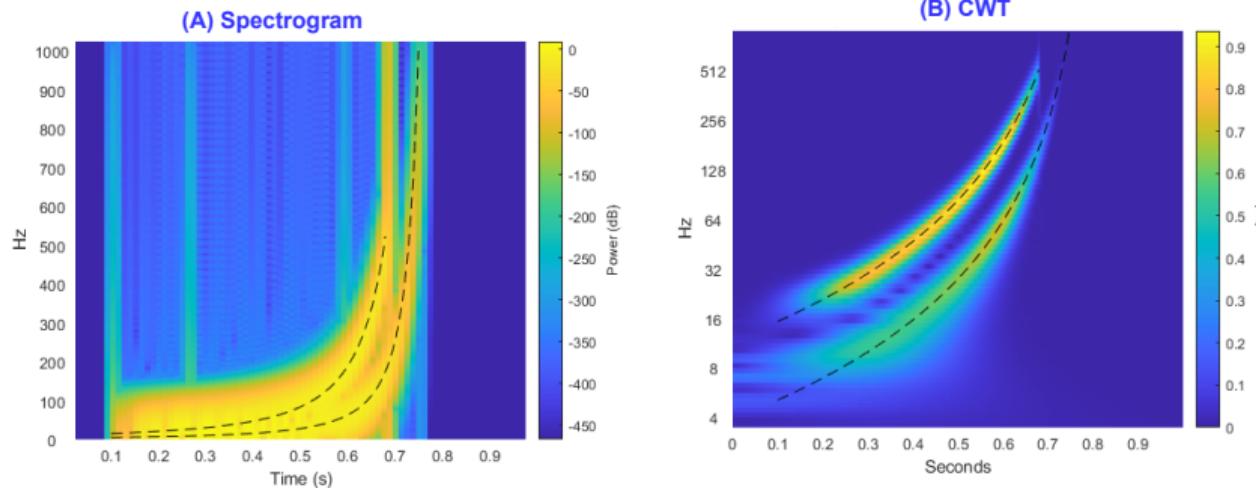


Figure: Difference in time-frequency resolution.



DEPARTMENT OF
ENGINEERING
SCIENCE



Thank you