FOUNDATION OF SENSOR SIGNAL PROCESSING (II)

FEATURE EXTRACTION IN TIME-FREQUENCY DOMAIN

Dr TIAN Jing

tianjing@nus.edu.sg

Module objective

Module: Time-frequency feature extraction for signal processing

Knowledge and understanding

- Understand the fundamentals of time-frequency domain signal representation and transformation, such as Fourier transformation and wavelet transformation

Key skills

- Design, build, implement and evaluate time-frequency feature extraction methods for signal processing

Major reference

- [Introduction] Steven W. Smith, ***The Scientist and Engineer's Guide to Digital Signal Processing***, available at http://www.dspguide.com
- [Practical] J. Unpingco, ***Python for Signal Processing: Featuring IPython Notebooks***, 2014, https://github.com/unpingco/Python-for-Signal-Processing
- [Practical] A. B. Downey, ***Think DSP: Digital Signal Processing in Python***, https://github.com/AllenDowney/ThinkDSP

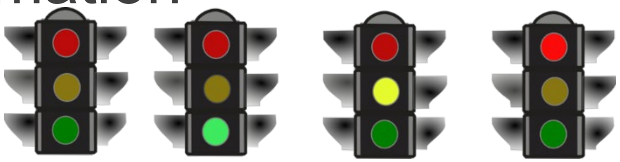
Topics

- Introduction to feature extraction in time-frequency domain of sensor signal
- Fourier transformation
- Wavelet transformation

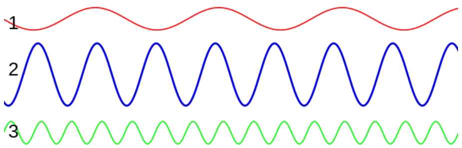
Intelligent sensing and sense making	
Physical Domain (our focus)	Business Domain
• Sensor-driven systems, e.g. robots, drones	• Non-sensor data, e.g. transactions, customer data • Also need sensor data (e.g., location, contact tracing, etc)

Recap: Signal

- A mechanism for conveying information
  - Gestures, traffic lights..



- Electrical engineering: Currents, voltages

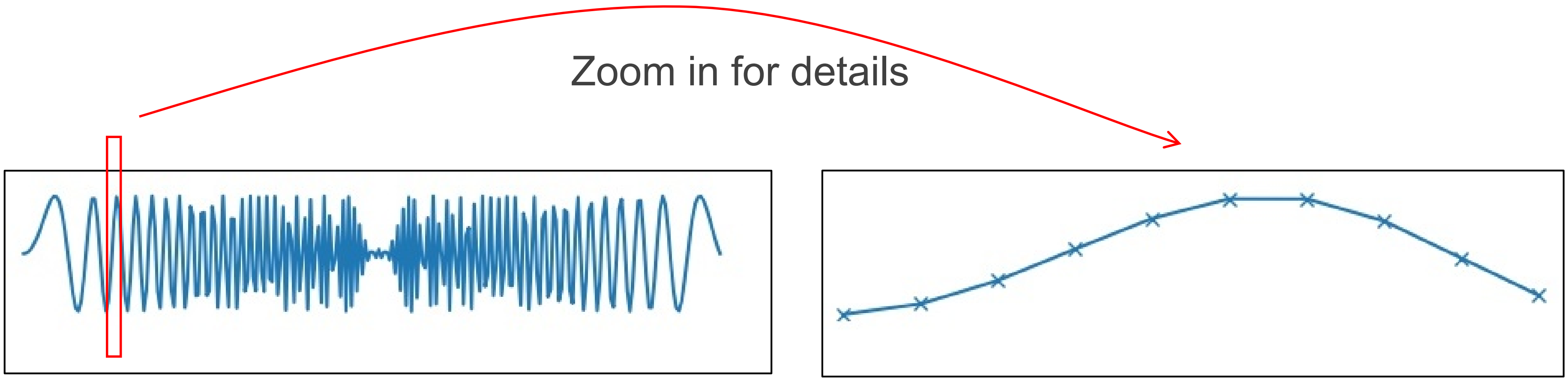


- Digital signals: Ordered collections of numbers that convey information, about a real world phenomenon, such as sounds, images

Source:
1. http://www.publicdomainfiles.com/show_file.php?id=13945761016913
2. https://commons.wikimedia.org/wiki/File:CPT-sound-pitchvolume.svg
3. https://commons.wikimedia.org/wiki/File:A)_Imagen_de_Lenna_en_escala_de_grises;_b)_Imagen_de_Lenna_con_el_filtro_de_Gauss_aplicado..jpg

Signal: Audio

- A sequence of numbers
  - The order in which the numbers occur is important
  - Represent a perceivable sound

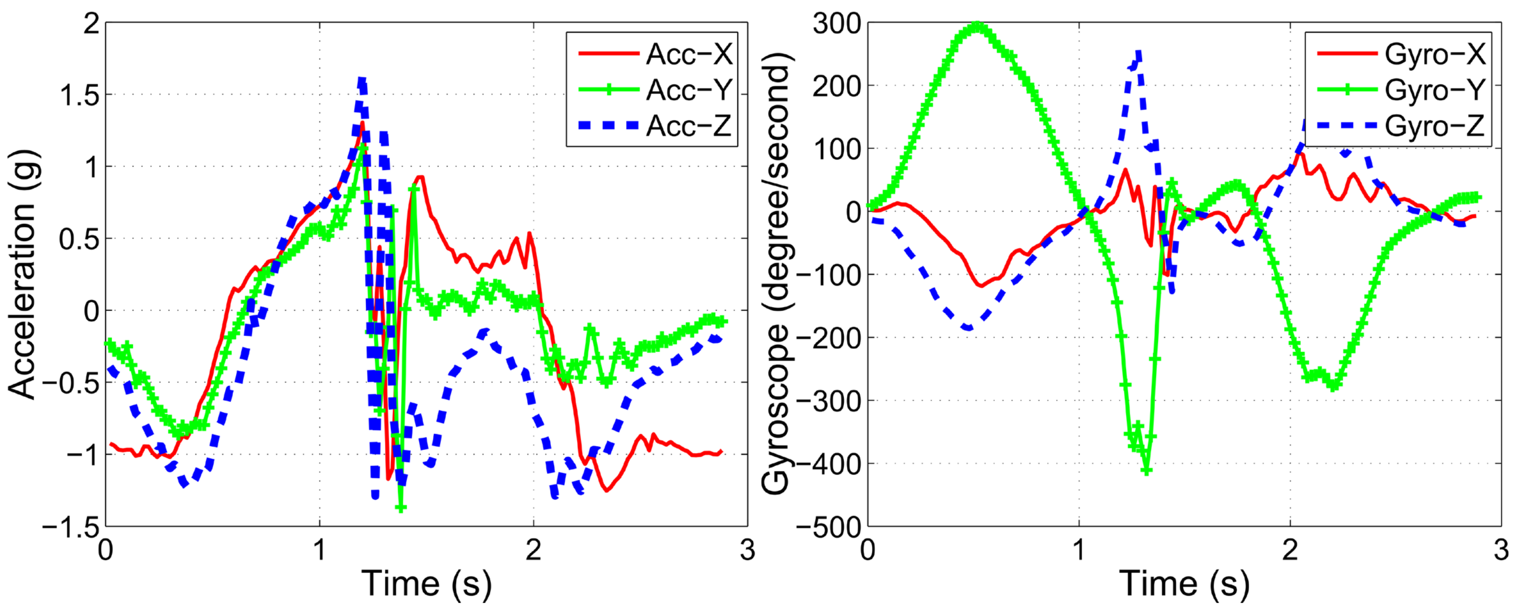


Signal: Image

- A rectangular arrangement (matrix) of numbers
  - Sets of numbers (for color images)
- Each pixel represents a visual representation of one of these numbers
  - E.g., 0 is minimum / black, 1 is maximum / white
  - Position / order is important

Signal: Wearable sensor signal

- Human activity classification using wearable sensory data.
- The UTD-MHAD dataset was collected using a wearable inertial sensor in an indoor environment. The dataset contains 27 actions performed by 8 subjects (4 females and 4 males). Each subject repeated each action 4 times. The inertial sensor signals were recorded using the inertial sensor signals (3-axis acceleration and 3-axis rotation signals).



Reference: https://personal.utdallas.edu/~kehtar/UTD-MHAD.html

Signal processing pipeline

Our focus

- **Sensor**
  - Various types of sensors, audio, vision, IoT, etc
- **Signal capture**
  - Sensing
  - Denoising
- **Signal transmission**
  - Source coding, channel coding, compression
- **Feature extraction**
  - Deterministic features
  - Data-driven features
  - Feature representation learning
- **Modeling**
  - Classification
  - Regression
  - Predication

Signal representation/Feature extraction (similar to feature engineering) uses mathematical transformations of raw input data to create new features to be used for further machine learning models.

- Use unstructured data sources
- Create features that are more easily interpreted
- Enhance creativity by using large sets of features

Reference: Chapter 5 Feature engineering, H. Brink, J. W. Richards, M. Fetherolf, Real-world machine learning, Manning, 2017.

Topics

- Introduction to feature extraction in time-frequency domain of sensor signal
- Fourier transformation
- Wavelet transformation

Time-frequency analysis of signal

One of top 10 algorithms in 20th century!

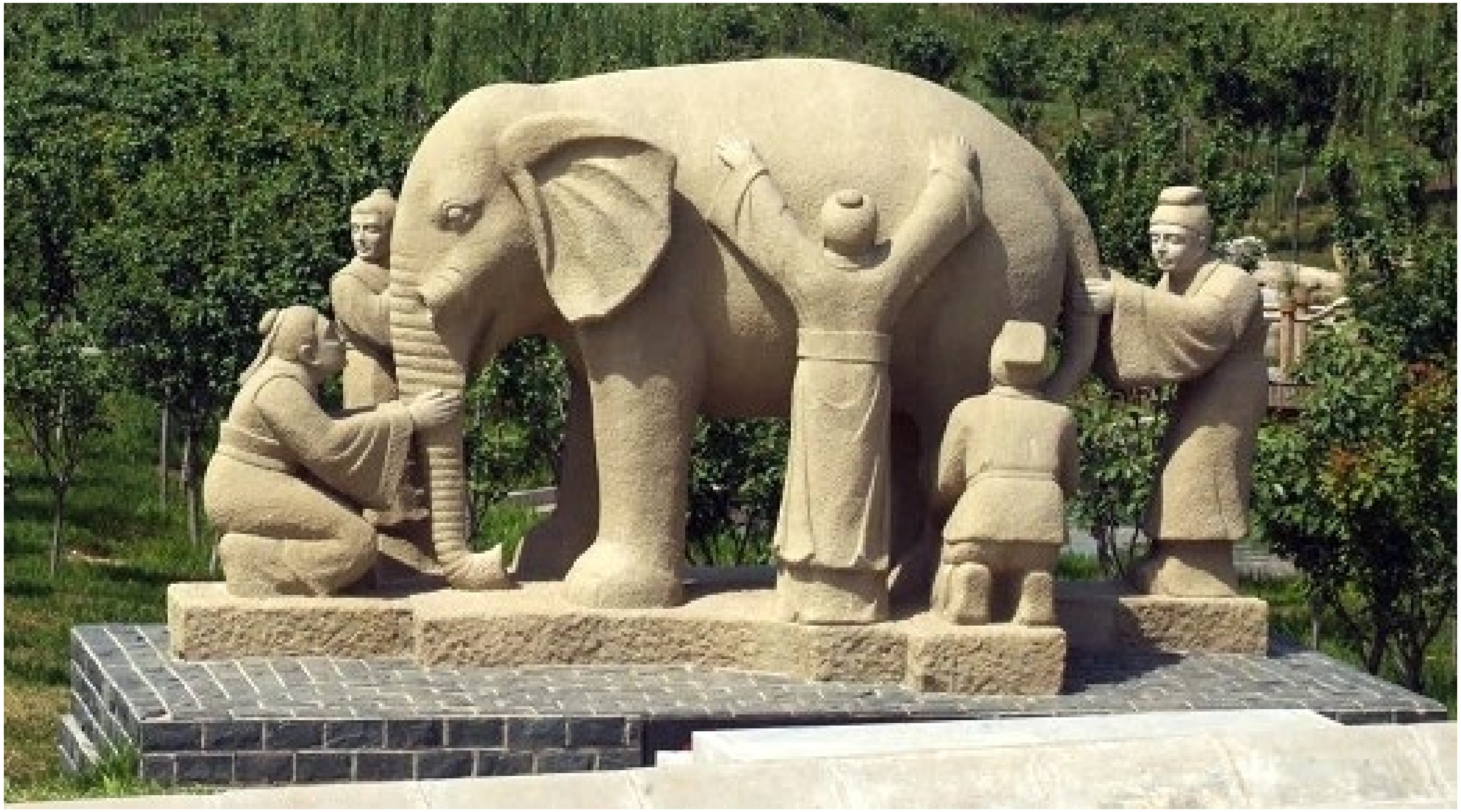
1. Metropolis algorithm for Monte Carlo
2. Simplex method for linear programming
3. Krylov subspace iteration
4. Decomposition approach to matrix computation (Singular value)
5. The Fortran compiler
6. QR algorithm for eigenvalues
7. Quick sort
8. Fast Fourier transform (FFT)
9. Integer relation detection
10. Fast multipole

Source: https://www.computer.org/csdl/mags/cs/2000/01/c1022.html

Signal decomposition

Intuition: Describe this image so that a listener can visualize what you are describing.

- Pixel-based descriptions are uninformative
- Content-based descriptions are infeasible in the general case



Source: https://cellcode.us/quotes/and-parable-blind-men-elephant.html

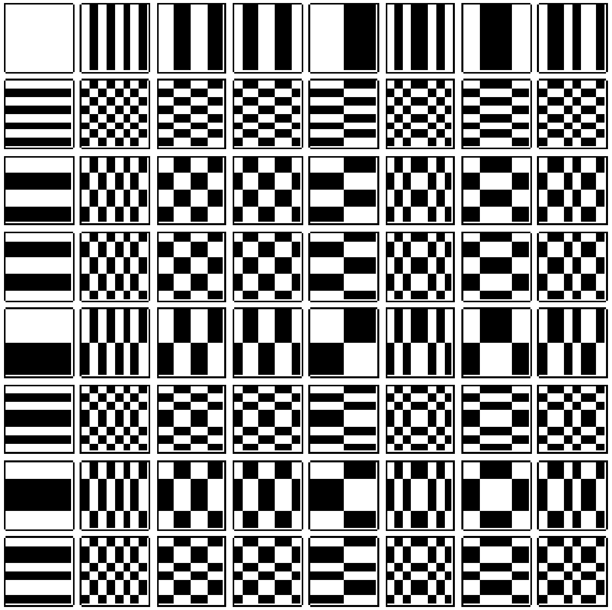
Signal decomposition

Objective: Decompose signal into 'basis', which is determined by certain frequency.

- Image signal decomposition: Checkerboard basis.
- Images have some slow/fast varying regions. For example, a first checkerboard picture with (slow varying) constant color. A second checkerboard picture that has fast changes.

Input image

A set of basis (checkerboard)



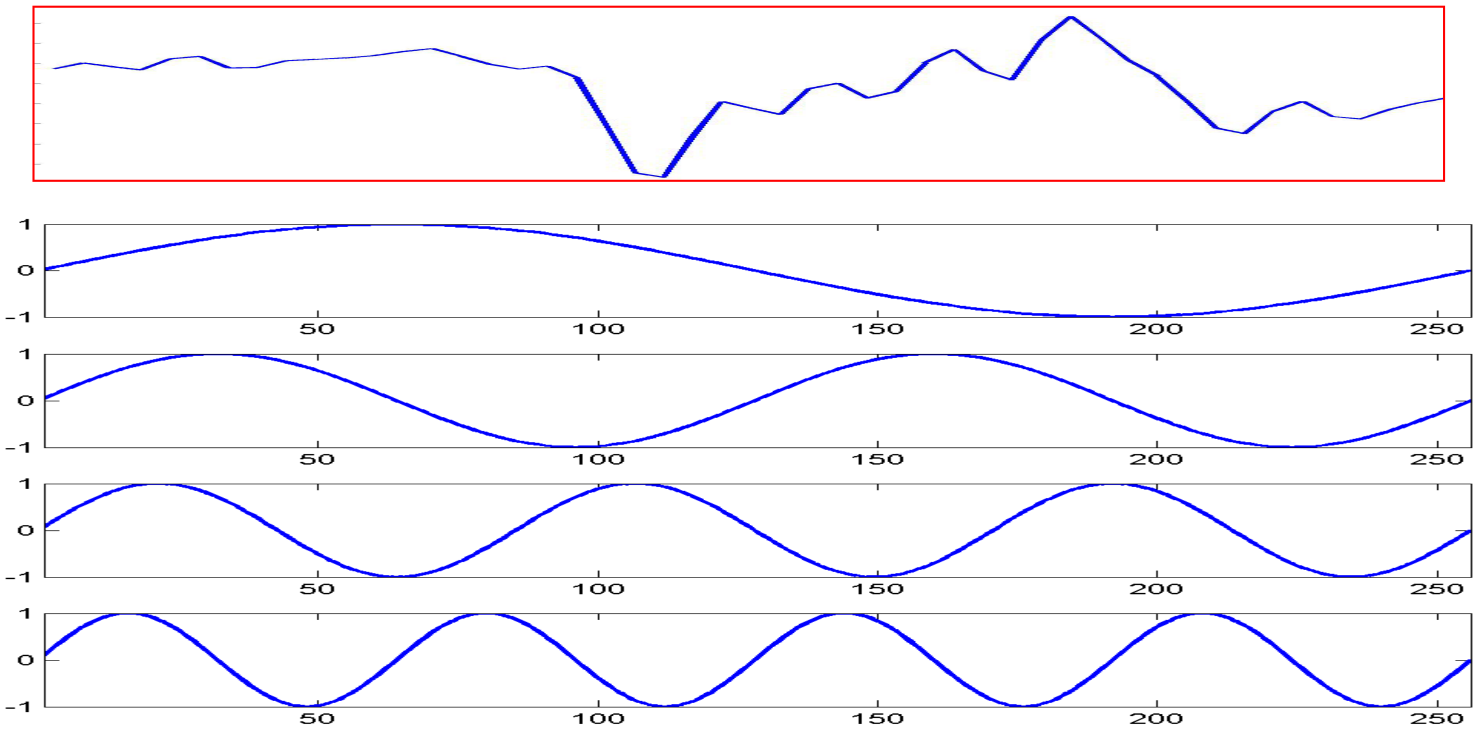
Note: The images are NOT visualized in the actual scale.

Reference: Hamamard basis image, https://en.wikipedia.org/wiki/Hadamard_transform

Signal decomposition

Objective: Decompose signal into ‘basis’, which is determined by certain frequency.

- Sound: Sinusoids basis.
- They are orthogonal.
- They can represent rounded shapes nicely.



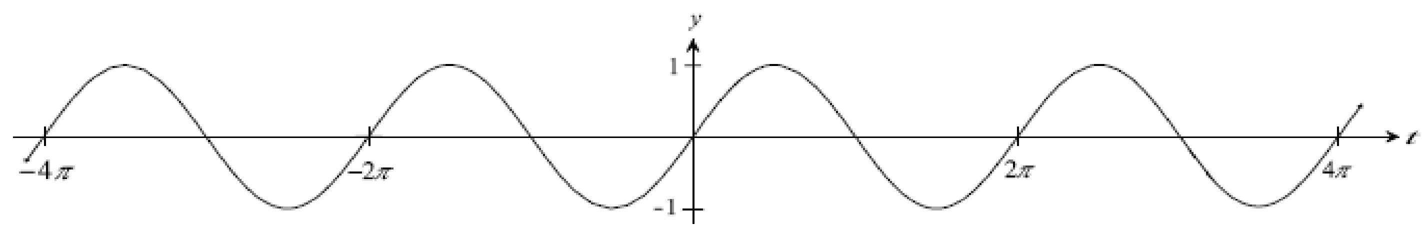
Sine and Cosine functions

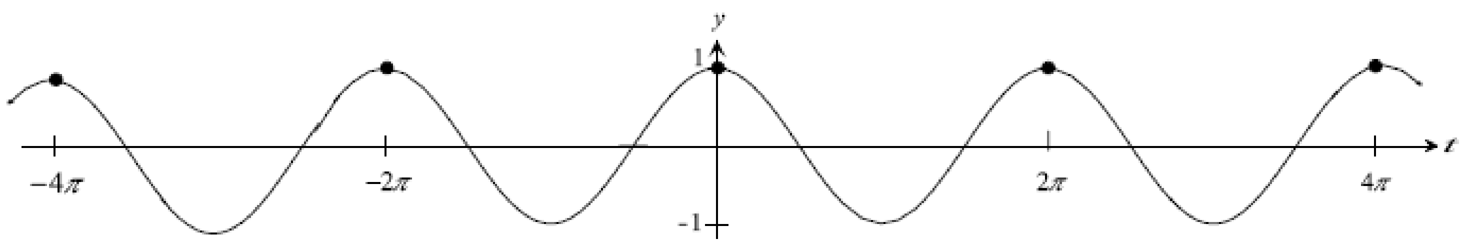
- Periodic functions
- General form of sine and cosine functions:

$\lvert A \rvert$	amplitude
$\dfrac{2\pi}{\lvert\alpha\rvert}$	period
b	phase shift

$$y(t) = A\sin(\alpha t + b) \qquad y(t) = A\cos(\alpha t + b)$$

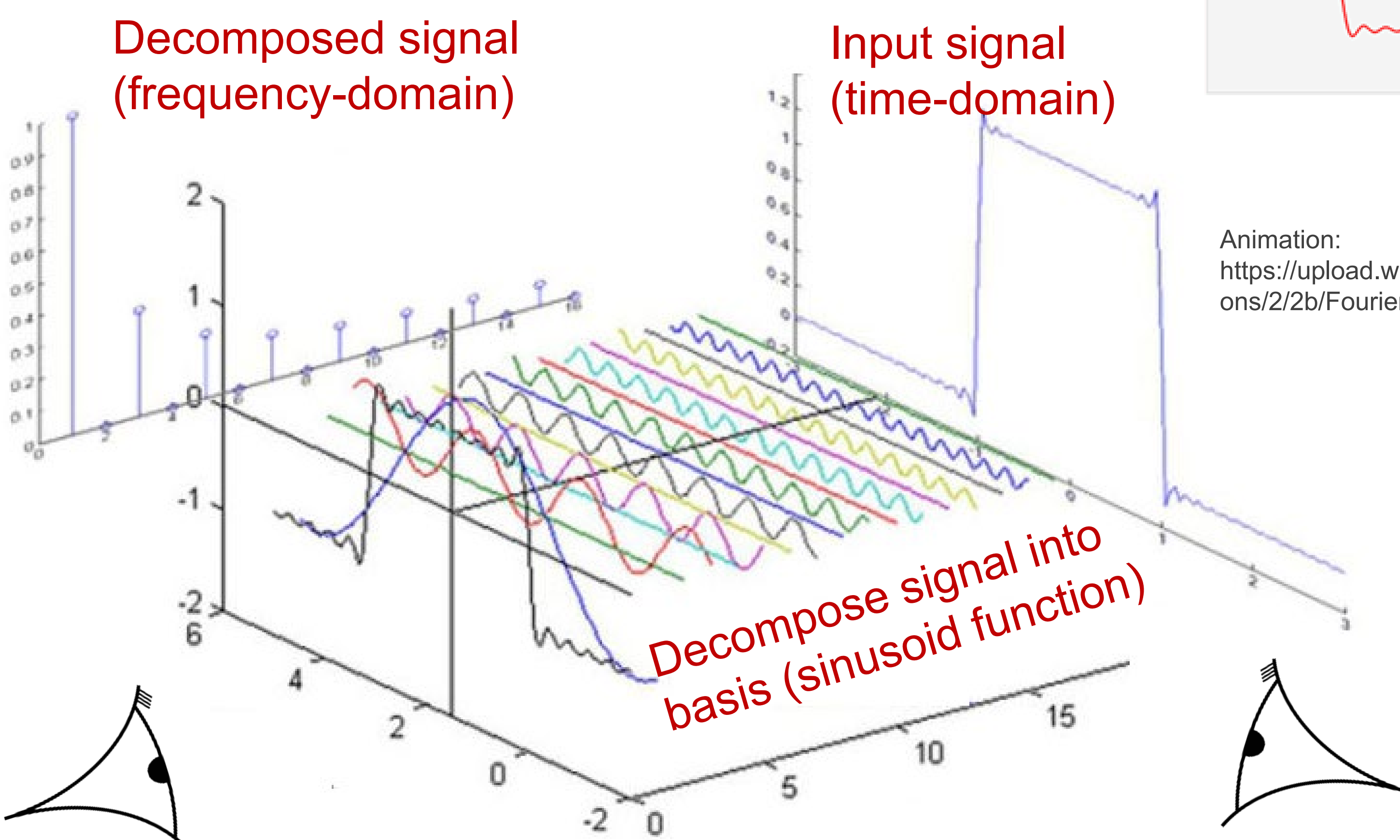
Example: $A = 1, b = 0, \alpha = 1$ period $= 2\pi$





Signal decomposition

Idea: Decompose the input signal from the time domain into a set of basis (that are sinusoid functions) to obtain the decomposed signal in the frequency domain.

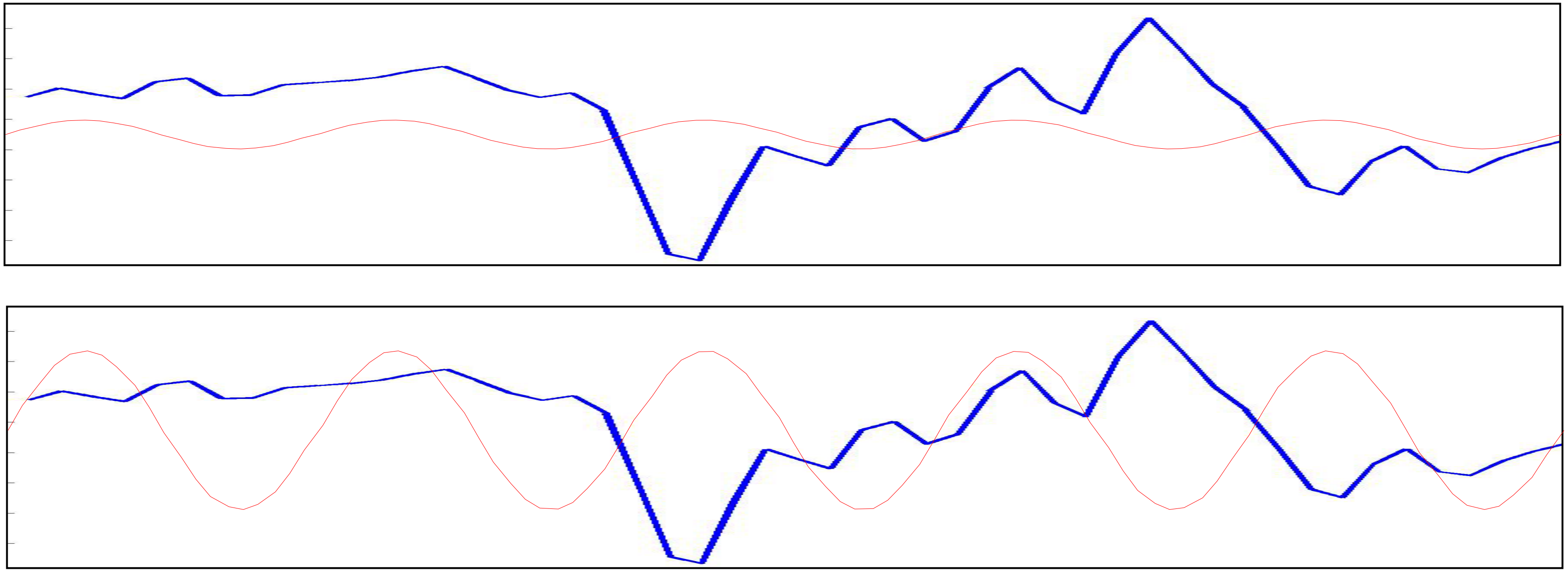


Animation:
https://upload.wikimedia.org/wikipedia/commons/2/2b/Fourier_series_and_transform.gif

View for time-domain

View for frequency-domain

Signal decomposition as optimization



Decompose signal (blue plot) into a set of sinusoid functions (red plot). Move the sinusoid left/right, and at each shift, adjust amplitudes. Find the combination of amplitude and phase that results in the smallest error (between red plot and blue dot) to fit the original signal.

Fourier analysis

Signal (Fourier domain) | Apply all signal values | Signal (time domain) | Basis (sinusoid functions)

$$\text{Forward: } F(u) = \sum_{x=0}^{N-1} f(x) e^{\frac{-i2\pi ux}{N}}, \text{where } u = 0,1,\cdots,N-1$$

$$\text{Inverse: } f(x) = \frac{1}{N} \sum_{u=0}^{N-1} F(u) e^{\frac{i2\pi ux}{N}}, \text{where } x = 0,1,\cdots,N-1$$

Note: $e^{ix} = \cos x + i \sin x$; $e^{i\pi} = \cos\pi + i\sin\pi = -1$, Reference: https://en.wikipedia.org/wiki/Euler%27s_identity

Example

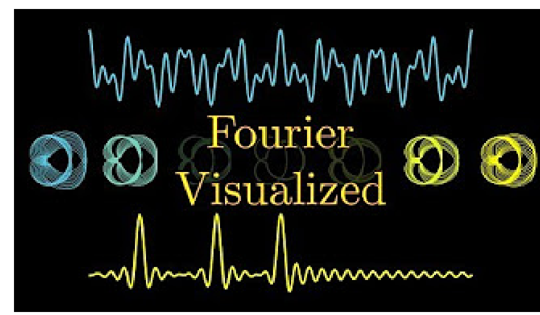
- Signal: $f(x) = [2, 3, 4, 4]$
- Fourier coefficients: $\mathrm{F}(u) = [13, (-2+i), -1, (-2-i)]$, where i is the imaginary unit

$$F(0) = \sum_{x=0}^{3} f(x) e^{\frac{-i2\pi 0x}{4}} = 2+3+4+4 = 13$$

$$F(1) = \sum_{x=0}^{3} f(x) e^{\frac{-i2\pi x}{4}} = 2e^{0} + 3e^{-i\pi/2} + 4e^{-i\pi} + 4e^{-i3\pi/2} = -2+i$$

$$F(2) = \sum_{x=0}^{3} f(x) e^{\frac{-i4\pi x}{4}} = 2e^{0} + 3e^{-i\pi} + 4e^{-i2\pi} + 4e^{-i3\pi} = -1$$

$$F(3) = \sum_{x=0}^{3} f(x) e^{\frac{-i6\pi x}{4}} = 2e^{0} + 3e^{-i3\pi/2} + 4e^{-i3\pi} + 4e^{-i9\pi/2} = -2-i$$

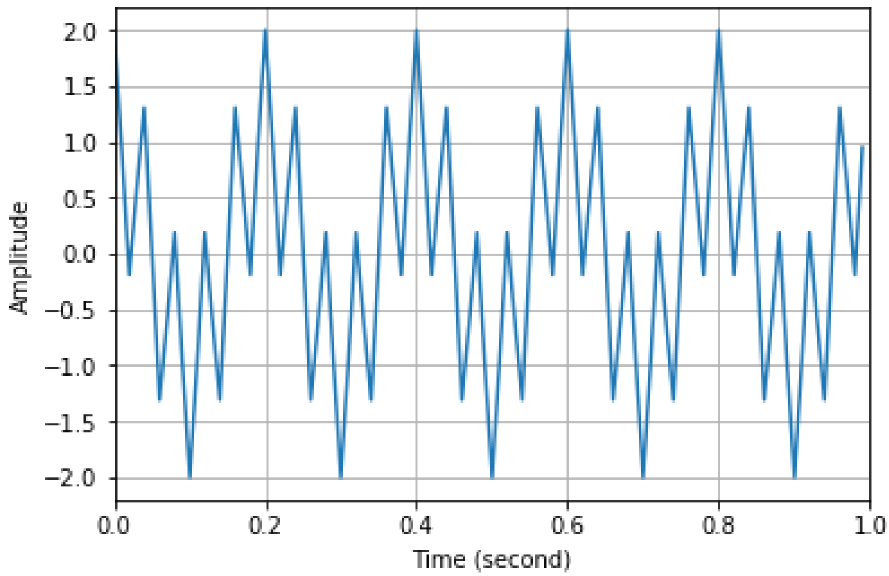


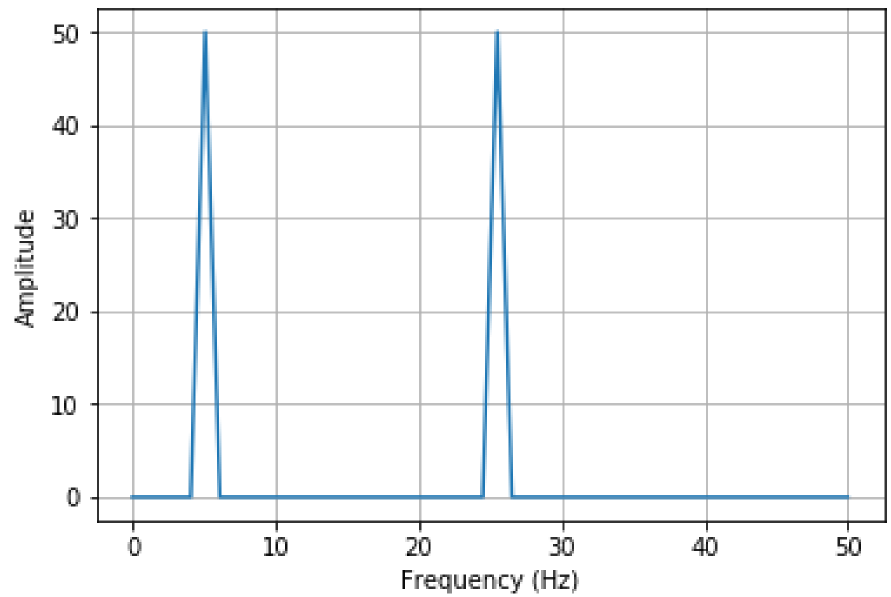
What is the Fourier Transform? A visual introduction
A 20-minute video tutorial on
https://www.youtube.com/watch?v=spUNpyF58BY
4.4 millions views since January 2018

Fourier analysis

Challenge: Provides good localization in the frequency domain but poor localization in the time domain. Has knowledge of what frequencies exist (example 1 in the current slide), but no information about where these frequencies are located in time (example 2 in the next slide).

Example 1: $f(t) = \cos(2\pi \cdot 5 \cdot t) + \cos(2\pi \cdot 25 \cdot t)$

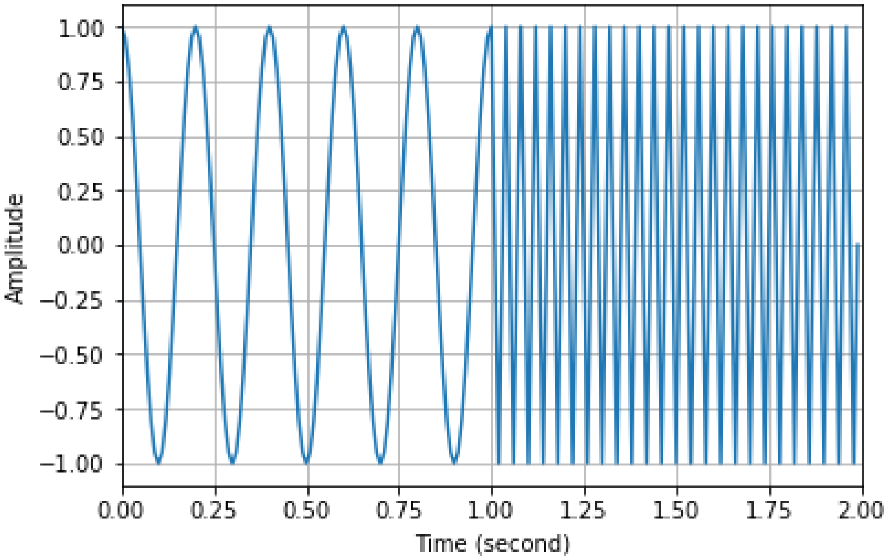


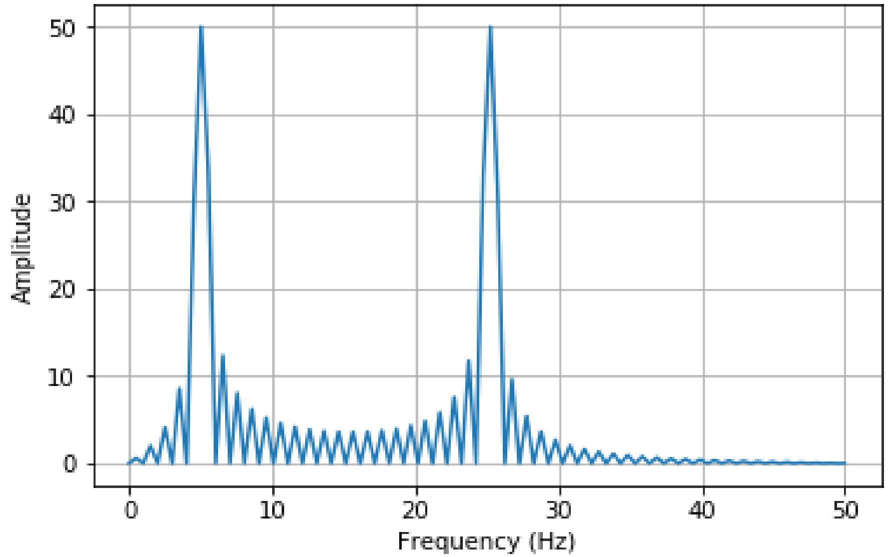


Fourier analysis

Challenge: Provides good localization in the frequency domain but poor localization in the time domain. Has knowledge of what frequencies exist (example 1 in the current slide), but no information about where these frequencies are located in time (example 2 in the next slide).

Example 2: $f(t) = \begin{cases} \cos(2\pi \cdot 5 \cdot t) & 0 \le t < 1 \\ \cos(2\pi \cdot 25 \cdot t) & 1 \le t < 2 \end{cases}$

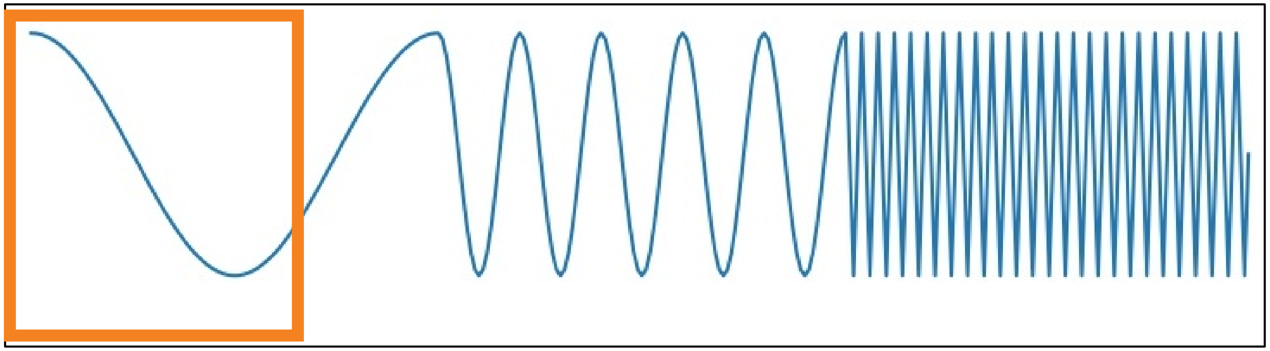




Short time Fourier transform

Idea: Segment the signal into narrow time intervals (i.e., narrow enough to be considered stationary) and take the Fourier transform of each segment.

- Choose a window function of finite length
- Place the window on start of the signal
- Truncate the signal using this window
- Compute Fourier transform of the truncated signal
- Incrementally slide the window to the right
- Repeat until window reaches the end of the signal

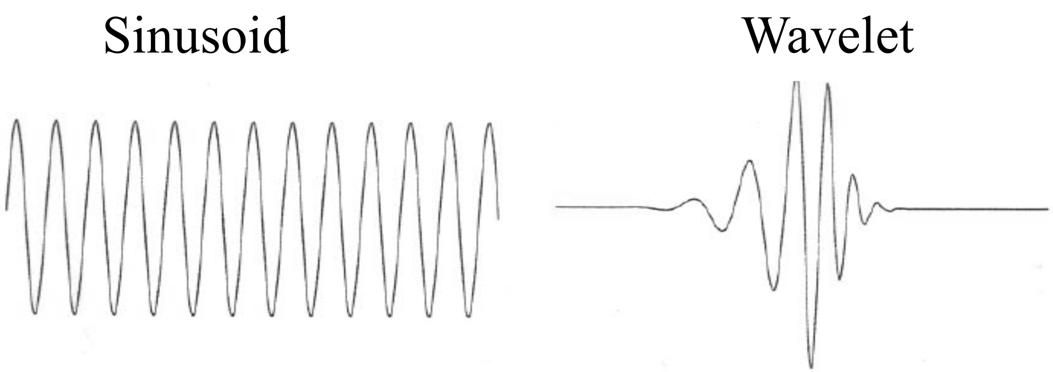


Topics

- Introduction to feature extraction in time-frequency domain of sensor signal
- Fourier transformation
- Wavelet transformation

What is a wavelet?

- A function that "waves" above and below the x-axis with the following properties
  - Varying frequency
  - Limited duration
- This is in contrast to sinusoids, used by Fourier transform, which have infinite duration and constant frequency.



Example 1: Decomposition

- Decompose signal by averaging and differencing the input signal to get approximation coefficients and detail coefficients, followed by downsampling.
- Note: Haar wavelet is used in this example. There are many other types of wavelet (see http://wavelets.pybytes.com/wavelet/haar/) in the literature.

input: [9 7 3 5]

Averaging filter [½, ½] with duplication padding

Differencing filter [½, -½] with duplication padding

$(9+7)/2=8$

[8 5 4 5]

[1 2 −1 0]

$(9-7)/2=1$

Down-sampling ratio 2

[8 4]

[1 −1]

Down-sampling ratio 2

Averaging filter [½, ½] with duplication padding

[6 4]

[2 0]

Differencing filter [½, -½] with duplication padding

Down-sampling ratio 2

[6]

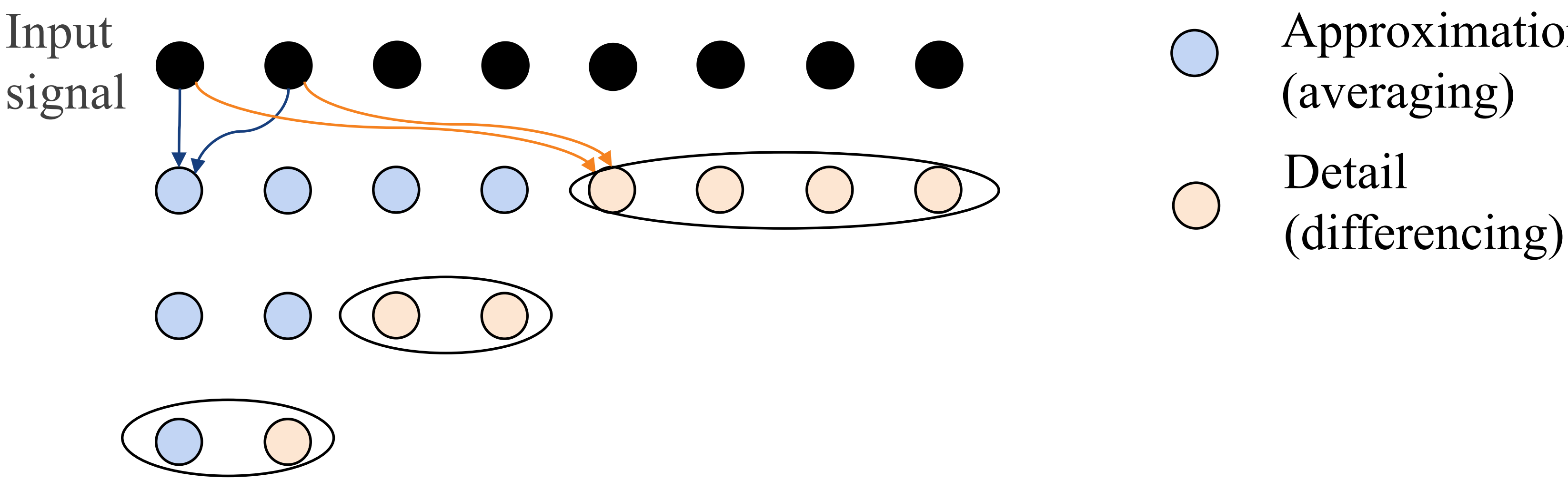
[2]

Down-sampling ratio 2

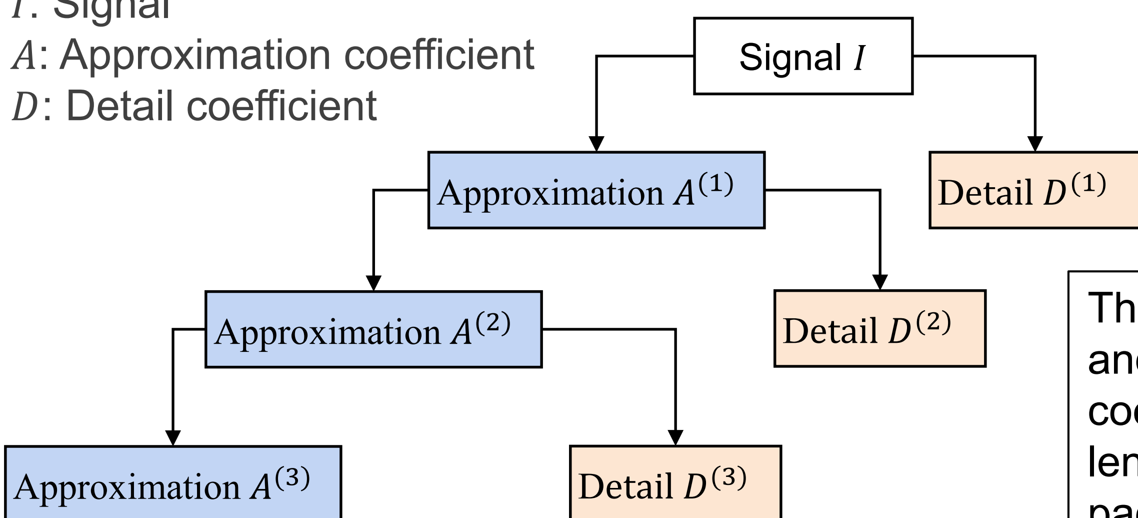
Result: Wavelet coefficients

[6 2 1 −1]

Overview: Signal decomposition



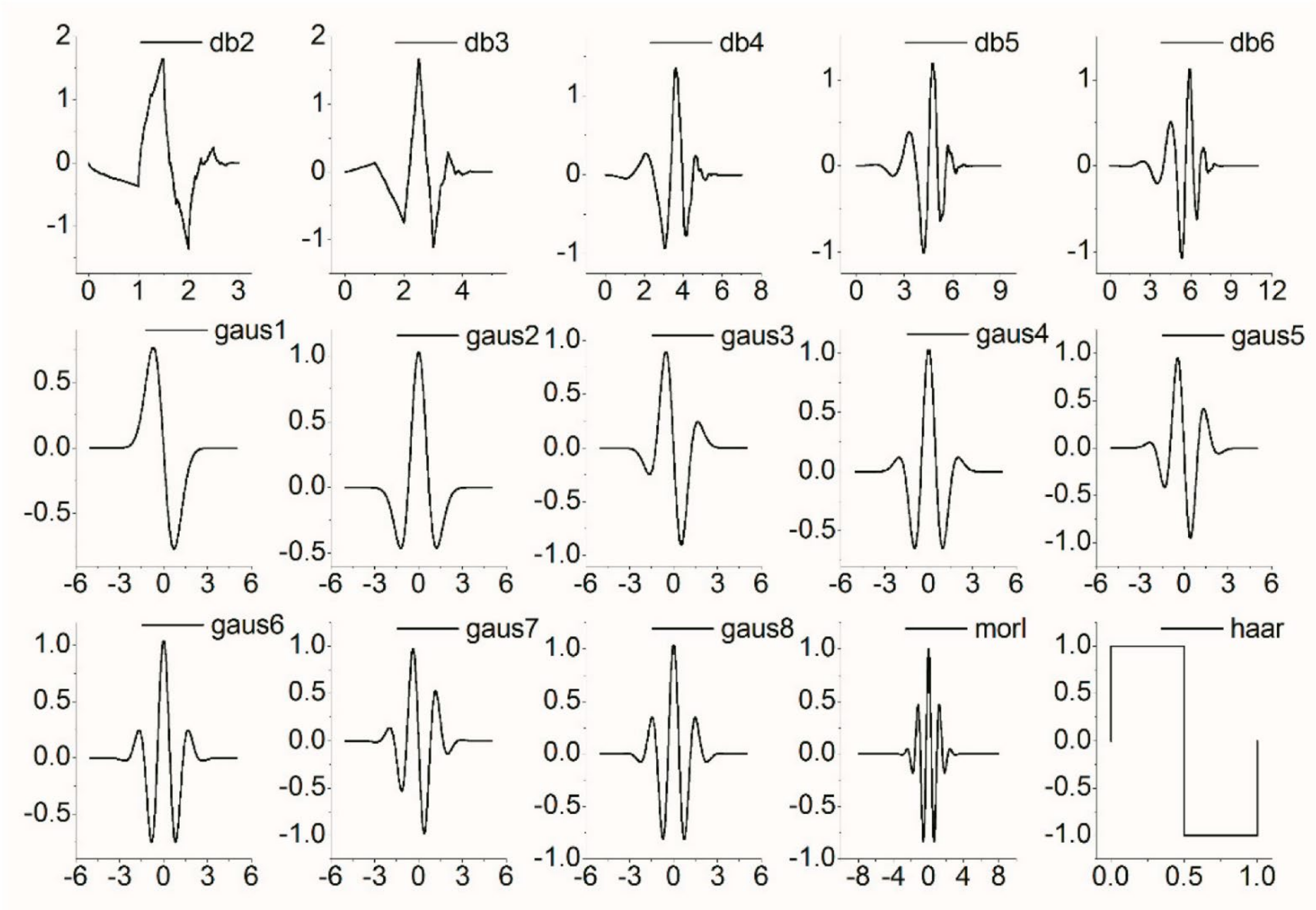
- I: Signal
- A: Approximation coefficient
- D: Detail coefficient



The maximum decomposition level and the dimension of output coefficients depend on (i) Signal length; (ii) Filter length; (iii) Signal padding method.

Wavelet: Other choice of filters

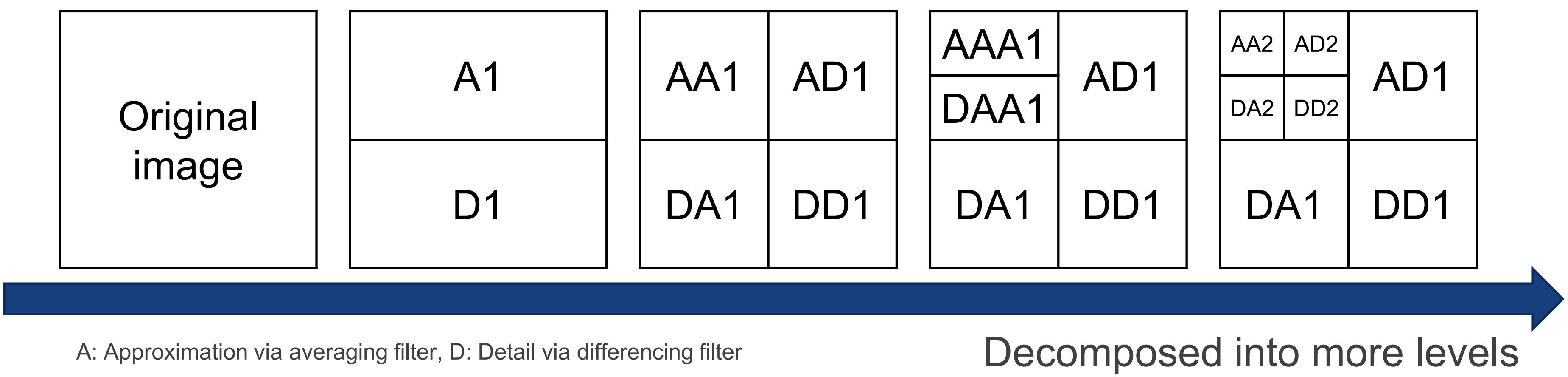
More choices: http://wavelets.pybytes.com/wavelet/haar/



Source: C. Xia and C. Liu, "Identification and Representation of Multi-Pulse Near-Fault Strong Ground Motion Using Adaptive Wavelet Transform," *Applied Sciences*, Vol. 9, No. 2, pp. 259, 2019 , https://www.mdpi.com/2076-3417/9/2/259

2D wavelet transformation

- AA: The upper left quadrant is filtered by the averaging filter along the rows and then filtered along the corresponding columns with the averaging filter. It represents the approximation coefficients of the original at half the resolution.
- DA/AD: The lower left and the upper right blocks are filtered along the rows and columns with averaging filter and differencing filter, alternatively. The AD block contains vertical edges. In contrast, the HL blocks shows horizontal edges.
- DD: The lower right quadrant is derived analogously to the upper left quadrant but with the use of the differencing filter, where we find edges of the original image in diagonal direction.



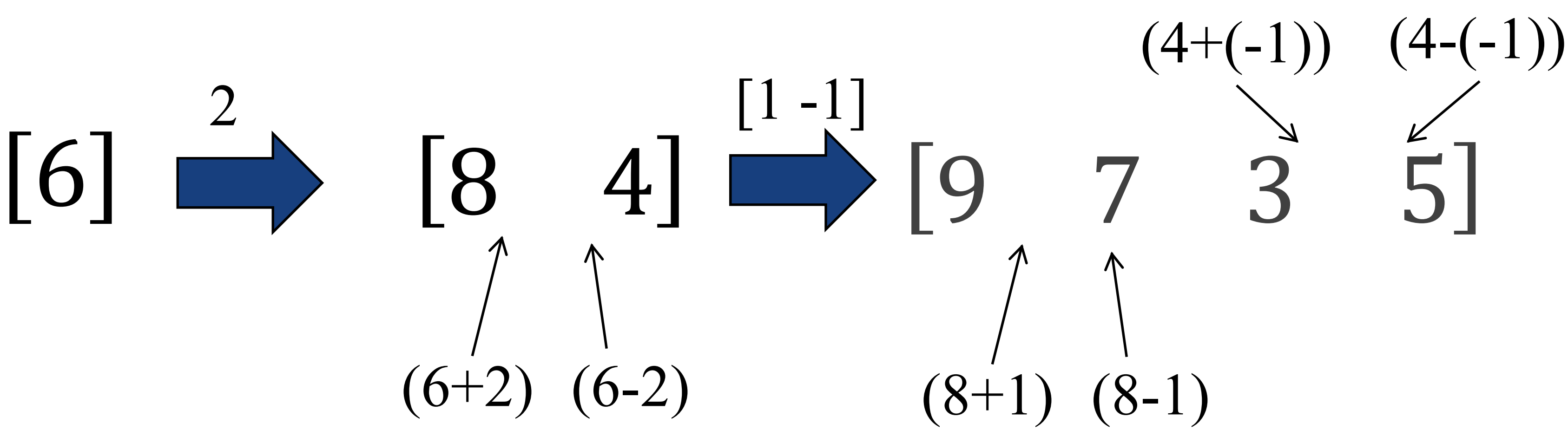
A: Approximation via averaging filter, D: Detail via differencing filter

Example: Reconstruction

- The original signal can be reconstructed by adding or subtracting the detail coefficients from the approximation coefficients.

Given the wavelet coefficients (obtained in previous example)

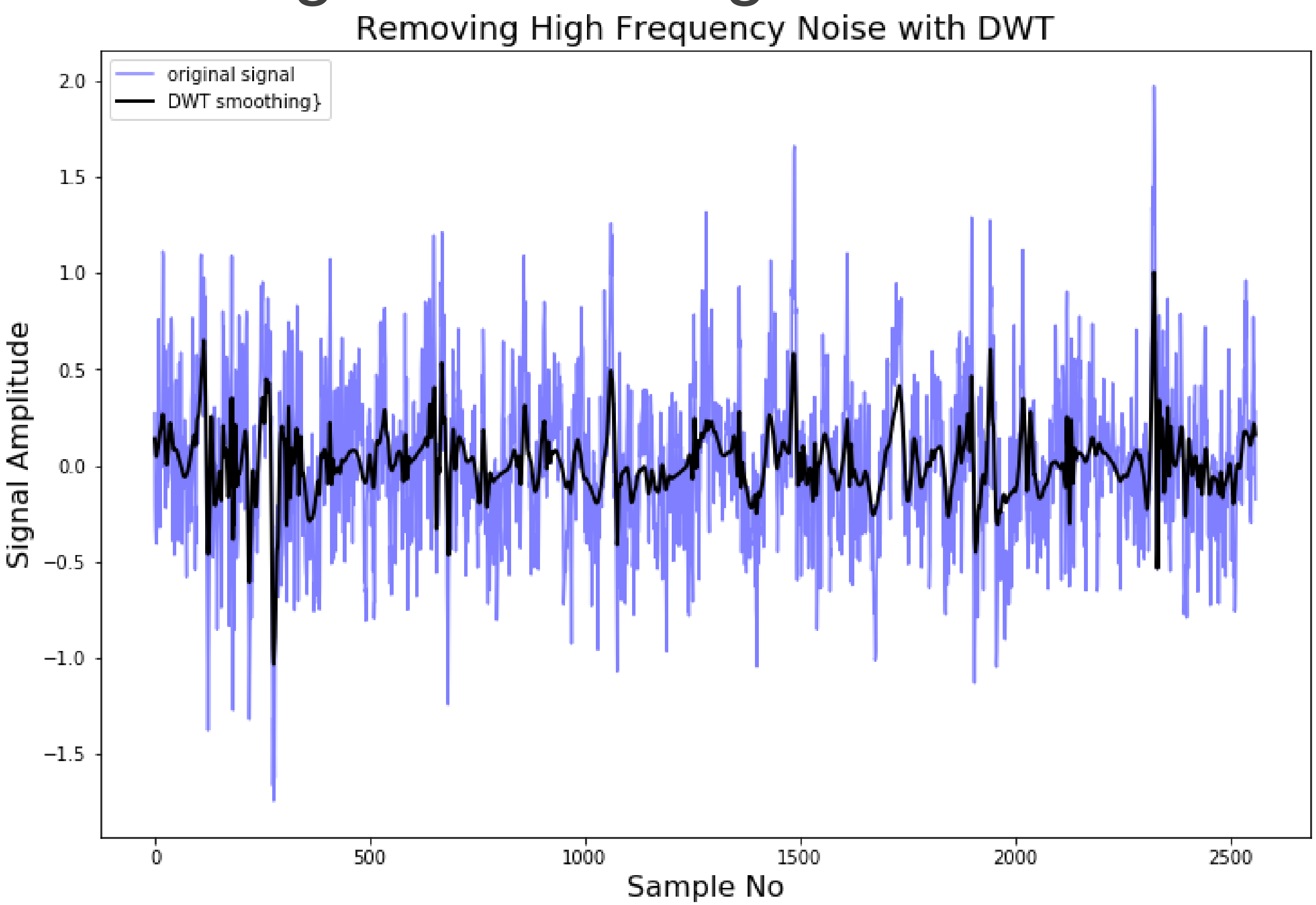
$$[6 \quad 2 \quad 1 \quad -1]$$



Note: This process is only applicable for the example in the previous slide. Other types of wavelet filters will have different calculation processes.

Wavelet application: Denoising

- Signal denoising



Signal

↓

Wavelet decomposition

↓

Manipulate wavelet coefficients

↓

Wavelet reconstruction

↓

Reconstructed signal

Reference: http://ataspinar.com/2018/12/21/a-guide-for-using-the-wavelet-transform-in-machine-learning/

Wavelet application: Feature extraction

- Perform wavelet transformation on signal
- Extract statistical features
  - Auto-regressive model coefficient values
  - Statistical features like variance, mean, median, etc.
- Perform signal classification or other machine learning tasks

Summary

- Introduction to feature extraction in time-frequency domain of sensor signal
- Fourier transformation
- Wavelet transformation

Thank you!

Dr TIAN Jing
Email: tianjing@nus.edu.sg