1 - Signals

Projection Theorem

- **I. The Fundamental Problem: Signal Approximation**
    - **Question:** Given a signal x in an N-dimensional space ($\mathbb{C}^N$), how can we find the **best possible approximation** of x using only a smaller set of K basis vectors (where $K < N$)?
    - This smaller set of K orthogonal basis vectors, u_k, spans a **subspace** $\mathcal{P}$.
    - "Best" is defined as the approximation $\tilde{x}$ that **minimizes the squared error** (or Euclidean distance) $||x - \tilde{x}||^2$.
- **II. The Projection Theorem**
    - **Statement:** The best approximation $\tilde{x}$ of a signal x in a subspace $\mathcal{P}$ is the **orthogonal projection** of x onto $\mathcal{P}$.
    - **The Formula:** This projection is constructed by summing the individual projections of x onto each of the K orthogonal basis vectors that span the subspace:

      $$\tilde{x} = \sum_{k=0}^{K-1} X_k u_k$$

      where the coefficients X_k are found by:

      $$X_k = \frac{(u_k, x)}{||u_k||^2} = \frac{u_k^H x}{u_k^H u_k}$$

    - **Key Property (Orthogonality of Error):** The error vector $e = x - \tilde{x}$ is **orthogonal** to the approximation $\tilde{x}$ and to every vector in the subspace $\mathcal{P}$.
    - **Geometric Interpretation:** Finding the best approximation is like "dropping a perpendicular" from the tip of the vector x onto the subspace $\mathcal{P}$. The point where it lands is $\tilde{x}$, and the perpendicular line itself is the error vector e.
- **III. How to Choose the "Best" Subspace?**
    - The theorem tells us how to project onto a *given* subspace, but how do we choose the K basis vectors that create the "best" subspace for approximation?
    - **Two Approaches:**
        1. **Fixed Basis (e.g., Fourier, DCT):** Start with a complete basis. Calculate all N expansion coefficients X_k. Then, select the K basis vectors corresponding to the K coefficients with the **largest magnitudes**. This subspace captures the most signal energy.
        2. **Signal-Dependent Basis (e.g., KLT/PCA):** Use techniques like Principal Component Analysis (PCA) to derive a **custom basis** tailored to the statistical properties of the signal class. The first K vectors of this basis are guaranteed to span the K-dimensional subspace that minimizes the mean square approximation error, making it the theoretical optimum.
- **IV. Consequences and Importance**
    - The Projection Theorem is a cornerstone concept with wide-ranging applications.

- **Data Compression:** It is the theoretical foundation for lossy compression. By keeping only the K most significant coefficients, we can represent the signal with fewer bits. The theorem guarantees this reconstruction has the minimum possible error for that K.
- **Noise Reduction:** If we assume the signal lies in a known subspace $\mathcal{P}$ and noise is spread across all dimensions, projecting the noisy signal onto $\mathcal{P}$ can effectively filter out the noise components that are orthogonal to the subspace.
- **Feature Extraction:** The coefficients X_k of the projection can be seen as the most significant **features** of the signal with respect to the chosen basis.
- **Basis for Advanced Algorithms:** The principle is fundamental to least squares estimation and PCA.

---

2 - Spectral Analysis

Time-Frequency Analysis: Spectrogram and Scalogram

- **I. The Problem: Limitations of Global Fourier Transform**
  - The standard Fourier Transform (FT) or DFT provides a **global** frequency representation.
  - It reveals **what** frequencies are present in a signal, but provides no information about **when** they occur.
  - This is a major drawback for **non-stationary signals** (like music or speech) where frequency content changes over time.

- **II. The Spectrogram (via Short-Time Fourier Transform - STFT)**
  - **Core Idea:** Analyze the frequency content of small, localized time segments of the signal.
  - **Process (STFT):**
    1. **Divide:** Split the signal into small, overlapping chunks of length N.
    2. **Window:** Apply a window function $w[n]$ to each chunk to reduce spectral leakage.
    3. **Transform:** Compute the DFT for each windowed chunk.
  - **STFT Formula:**

$$S[k, m] = \sum_{n=0}^{N-1} x[mM + n]w[n]e^{-j2\pi nk/N}$$

    where m is the time-chunk index and k is the frequency index.
  - **The Spectrogram:** It is the **visual representation of the squared magnitude** of the STFT coefficients: $|S[k, m]|^2$. It's a 2D plot showing energy distribution across time (x-axis) and frequency (y-axis).
  - **Fundamental Limitation: The Time-Frequency Trade-off (Uncertainty Principle)**
    - **Time Resolution** is determined by the window size N. A short window (small N) gives good time resolution.
    - **Frequency Resolution** is F_s/N. A long window (large N) gives good frequency resolution.
    - **The Trade-off:** You cannot simultaneously have high resolution in both time and frequency. The STFT uses a **fixed window size**, meaning the resolution is the same for all frequencies.

- **III. The Scalogram (via Continuous Wavelet Transform - CWT)**
  - **Motivation:** To overcome the fixed-resolution limitation of the STFT.
  - **Core Idea: Multi-Resolution Analysis**.
    - Use **short windows** (compressed wavelets) for **high frequencies** to get good time resolution.
    - Use **long windows** (stretched wavelets) for **low frequencies** to get good frequency resolution.
  - **Process (CWT):**
    1. **Mother Wavelet $h(t)$:** A prototype function that is localized in both time and frequency.
    2. **Wavelet Family $h_{a,\tau}(t)$:** Generated by scaling (by a) and translating (by τ) the mother wavelet.
    3. **The CWT:** An inner product that measures the similarity between the signal $x(t)$ and the wavelet at a specific scale a and time τ.
  - **The Scalogram:** The squared magnitude of the CWT coefficients: $|CWT(\tau, a)|^2$. It describes how the signal's energy is distributed over the **time-scale plane**.

- **IV. Comparison: Spectrogram vs. Scalogram**
  - **Time-Frequency Tiling:** This is the key difference, visualized in the diagrams.
    - **Spectrogram (STFT):** Has a **uniform tiling** of the time-frequency plane. Resolution is fixed for all frequencies.
    - **Scalogram (CWT):** Has an **adaptive, non-uniform tiling**.
      - At high frequencies: Tiles are narrow in time and wide in frequency (good time res, poor freq res).
      - At low frequencies: Tiles are wide in time and narrow in frequency (poor time res, good freq res).
  - **Conclusion:** The Scalogram's adaptive analysis is better suited for signals with both transient bursts and long, stationary components.

Comparing DTFT and DFT

- **I. Introduction**
  - Both the Discrete-Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT) are tools for analyzing the frequency content of discrete-time signals.
  - The DTFT is a theoretical tool, while the DFT is a practical, computable tool that can be seen as a sampled version of the DTFT.

- **II. Definitions and Formulas**
  - **Discrete-Time Fourier Transform (DTFT):**
    - **Formula:**

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$$

    - **Input:** An **infinite-length** (or zero-padded) discrete-time sequence $x[n]$.

- **Output:** $X(e^{j\omega})$, a **continuous and periodic function** of the normalized angular frequency ω. The period is 2π.
- **Discrete Fourier Transform (DFT):**
  - **Formula:**

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N}$$

  - **Input:** A **finite-length** N-point discrete-time sequence $x[n]$.
  - **Output:** $X[k]$, a **discrete sequence** of N frequency coefficients.

- **III. Key Differences (Summary Table)**

Feature	Discrete-Time Fourier Transform (DTFT)	Discrete Fourier Transform (DFT)
Input Signal	Infinite or finite discrete-time sequence $x[n]$	Finite-length (N-point) sequence $x[n]$
Time Domain	Generally Aperiodic	**Implicitly Periodic** with period N
Frequency Domain	**Continuous** function of frequency	**Discrete** sequence of N frequency samples
Spectrum Period	Periodic with period 2π (for ω) or F_s (for f)	Periodic with period N (for index k)
Computability	Theoretical tool; not directly computed in its entirety	Directly computable (via **Fast Fourier Transform - FFT**)
Represents	The "true" spectrum of a discrete-time sequence	Samples of one period of the DTFT of the N-point sequence

- **IV. Relationship through Sampling and Periodization**
  - The DFT is fundamentally linked to the DTFT through the operations of sampling and periodization.
  - **The Bridge:** The N coefficients of the DFT, $X[k]$, are exactly equal to N samples of one period of the DTFT, $X(e^{j\omega})$, evaluated at the discrete frequencies $\omega = 2\pi k/N$. $X[k] = X(e^{j\omega})|_{\omega=2\pi k/N}$
  - **Implicit Periodicity:** The DFT operates on a finite N-point sequence $x[n]$. However, because its basis functions ($e^{j2\pi kn/N}$) are periodic with period N, the DFT inherently treats the input signal $x[n]$ as if it were one period of an infinitely periodic signal.
    - **Consequence 1 (Time Domain):** This assumed periodicity is what leads to **circular convolution** when multiplying DFTs.
    - **Consequence 2 (Frequency Domain):** Treating the time-domain signal as periodic is equivalent to **sampling** its continuous spectrum (the DTFT). This is why the DFT output $X[k]$ is a discrete set of frequency samples.

- **V. Conclusion**

- The DTFT provides the complete, continuous spectrum of a discrete-time sequence, serving as a vital theoretical concept. The DFT provides a discrete, finite, and computable set of samples of that spectrum, making it the practical workhorse for digital spectral analysis, implemented efficiently via the FFT algorithm.

Linking Continuous and Discrete Time Spectra: Sampling and Periodization

- **I. Introduction**
  - The relationship between the continuous-time Fourier Transform (FT) and the Discrete Fourier Transform (DFT) is not arbitrary. It is governed by the dual operations of **sampling** and **periodization**.
  - A fundamental duality exists: what happens in the time domain has a corresponding, inverse effect in the frequency domain.
- **II. The Sampling Property: Time Sampling → Frequency Periodization**
  - **Operation (Time Domain):** A continuous-time signal $x(t)$ is **sampled** at a rate of $F_s = 1/T$, creating a discrete sequence of impulse strengths $x(kT)$.
  - **Effect (Frequency Domain):** The spectrum of the sampled signal, $X_s(f)$, becomes a **periodic replication** of the original continuous spectrum $X(f)$. The replicas are centered at integer multiples of the sampling frequency F_s.
  - **Formula:**

$$X_s(f) = F_s \sum_{k=-\infty}^{\infty} X(f - kF_s)$$

  - **Mantra: Sampling in the time domain leads to periodization in the frequency domain**.
  - **Consequence:** This is the foundation of the **Nyquist-Shannon Sampling Theorem**. If F_s is high enough ($\geq 2B$), the replicas don't overlap (no aliasing), and the original signal can be recovered.
- **III. The Periodization Property: Time Periodization → Frequency Discretization**
  - **Operation (Time Domain):** A finite-support (or single period) signal $g(t)$ is made **periodic** by summing infinite, shifted copies of itself with a period T.
  - **Effect (Frequency Domain):** The spectrum of the resulting periodic signal $x(t)$ is no longer continuous. It becomes a **discrete spectrum**, consisting of a series of impulses (Dirac deltas).
  - The impulses are located at discrete frequencies $f = k/T$, which are integer multiples (harmonics) of the fundamental frequency $1/T$.
  - The amplitude of each impulse is proportional to the value of the original continuous spectrum $G(f)$ sampled at that harmonic frequency: $\frac{1}{T}G(k/T)$.
  - **Mantra: Periodization in the time domain leads to discretization (sampling) in the frequency domain**.
  - **Consequence:** This is the mathematical basis for **Fourier Series**, which represents a periodic signal as a sum of discrete frequency components (harmonics).
- **IV. Synthesis: The Four Fourier Representations**

- The interplay between these two properties is key to understanding the different Fourier transforms.
  1. **Continuous & Aperiodic (FT):** A continuous, aperiodic signal $s(t)$ has a continuous, aperiodic spectrum $S(f)$.
  2. **Continuous & Periodic (Fourier Series):** Making $s(t)$ periodic in time **discretizes** its spectrum into harmonics $S(k)$.
  3. **Discrete & Aperiodic (DTFT):** Sampling $s(t)$ in time makes its spectrum **periodic**, $S_{1/T}(f)$.
  4. **Discrete & Periodic (DFT):** The DFT operates on a finite N-point sequence, which is **implicitly periodic**. Therefore, its spectrum is both **discrete** (from the time periodicity) and **periodic** (from the time sampling).

- **V. Conclusion**

  - The DFT, the main tool for digital spectral analysis, can be understood as the result of applying both sampling and periodization to a continuous signal. The sampling operation makes the spectrum periodic, and the implicit periodicity of the DFT operation discretizes that periodic spectrum, resulting in the finite set of N coefficients we compute.

3 - Data Compression

Block Transform Coding for Data Compression

- **I. Motivation for Compression**

  - **Problem:** Uncompressed signals, especially images and video, require massive amounts of storage. A 2-hour movie could be ~224 GB.
  - **Solution: Source Encoding (Compression).** This is possible because signals contain **redundancy**.
    - **Spatial Redundancy:** Adjacent pixels are often similar.
    - **Irrelevant Information:** Data that is imperceptible to human senses.
  - **Goal:** Reduce the number of bits (b') needed to represent the signal compared to the original (b), measured by the compression ratio $c = b/b'$.

- **II. The General Compression Framework**

  - **Encoder:** `Input -> Mapper -> Quantizer -> Symbol Coder -> Compressed Data`.
    - **Mapper:** Transforms data to a new format to reduce redundancy (e.g., decorrelate data).
    - **Quantizer:** Reduces the precision of the data, which is the primary source of **lossy compression**. It discards less important information.
    - **Symbol Coder:** Assigns codes (often variable-length) to the quantized data.
  - **Decoder:** Reverses the process: `Symbol Decoder -> Inverse Mapper -> Reconstructed Image`.

- **III. Block Transform Coding: A Specific Method**

  - This is a popular method that fits the general framework, with JPEG being a prime example.
  - **Core Idea:** Divide the image into small blocks (e.g., 8×8 pixels) and process each independently.

- **Step-by-Step Process (Encoder):**
  1. **Image Division:** The input image is divided into $n \times n$ subimages (blocks).
  2. **Forward Transform (The Mapper):** A transform like the **Discrete Cosine Transform (DCT)** is applied to each block.
     - **Goal:** This transform compacts the energy of the block into a few coefficients (mostly low-frequency). The $T(0, 0)$ or DC coefficient represents the average intensity, while others (AC coefficients) represent details.
  3. **Quantization:** This is the crucial lossy step. Each transform coefficient $T(u, v)$ is divided by a value from a **quantization table** $Z(u, v)$ and rounded.
     - $Z(u, v)$ has larger values for high-frequency coefficients, quantizing them more coarsely (discarding fine details).
  4. **Symbol Encoding:** The quantized coefficients are encoded.
     - **Zigzag Scan:** The 2D block of coefficients is reordered into a 1D sequence using a zigzag pattern. This groups the more significant low-frequency coefficients first, followed by long runs of zeros.
     - **Coding:** Run-Length Encoding (RLE) is used for the runs of zeros, and Huffman or arithmetic coding is used for the remaining values.

- **IV. Zonal vs. Threshold Coding (Quantization Strategies)**

  - **Zonal Coding:**
    - **Principle:** Assumes the most important information (high variance coefficients) is always in a fixed, predefined "zone," typically the low-frequency region.
    - **Implementation:** A **zonal mask** is used to keep coefficients inside the zone and discard those outside.
  - **Threshold Coding:**
    - **Principle:** Assumes the most important coefficients are those with the **largest magnitudes**, regardless of their location.
    - **Implementation:** A threshold is set. Coefficients with magnitudes above the threshold are kept; those below are discarded.
    - **Advantage:** More adaptive to block content. If a block has important high-frequency details (like an edge), this method can preserve them. JPEG's quantization table method is a form of location-dependent thresholding.

- **V. Conclusion**

  - Block Transform Coding is an effective and widely used compression strategy. It works by transforming spatial data into a frequency domain where energy is compacted, allowing for aggressive but perceptually-guided information removal through quantization, followed by efficient lossless coding.

---

4 - Linear Processing

Characterization and Analysis of LTI Systems

- **I. System Definition and Key Properties**

  - A **system** transforms an input signal $x(t)$ into an output signal $y(t)$.

  - **Linear Time-Invariant (LTI)** systems are a crucial class because they are easy to analyze. They must satisfy two properties:
    1. **Linearity:** The system obeys the superposition principle. The response to a weighted sum of inputs is the weighted sum of the individual responses. $\mathcal{L}(a_1x_1(t) + a_2x_2(t)) = a_1\mathcal{L}(x_1(t)) + a_2\mathcal{L}(x_2(t))$.
    2. **Time-Invariance:** The system's behavior does not change over time. A time-shifted input $x(t - t_0)$ produces a correspondingly time-shifted output $y(t - t_0)$.

- **II. Time-Domain Characterization**

  - **Impulse Response $h(t)$:** An LTI system is **completely characterized** in the time domain by its impulse response.
  - **Definition:** $h(t)$ is the output of the system when the input is a Dirac delta function $\delta(t)$. $h(t) = \mathcal{L}\delta(t)$.
  - **Convolution:** The output $y(t)$ for *any* arbitrary input $x(t)$ is found by **convolving** the input with the system's impulse response.
    - **Formula:**

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau$$

  - **Significance:** Knowing $h(t)$ provides a complete external description of the system's behavior, without needing to know its internal structure.

- **III. Frequency-Domain Characterization**

  - **Frequency Response $H(f)$:** An LTI system is **completely characterized** in the frequency domain by its frequency response.
  - **Definition:** $H(f)$ is the **Fourier Transform of the impulse response** $h(t)$. $H(f) = \mathcal{F}h(t)$.
  - **Input-Output Relationship:** The Fourier transform of the output $Y(f)$ is the **product** of the Fourier transform of the input $X(f)$ and the system's frequency response $H(f)$.
    - **Formula:** $Y(f) = H(f)X(f)$.
  - **The Convolution Property:** This simple multiplication in frequency is a direct consequence of the convolution property of the Fourier Transform: **convolution in the time domain becomes multiplication in the frequency domain**.
  - **Interpretation of $H(f)$:**
    - **Magnitude $|H(f)|$:** Represents the gain of the system. It shows how much the system **amplifies or attenuates** each frequency component f.
    - **Phase $\angle H(f)$:** Represents the **phase shift** the system introduces at each frequency f.

- **IV. Application: Filters**

  - Filters are a primary application of LTI systems, designed to selectively pass or block frequencies. Their behavior is defined by their frequency response $H(f)$.
  - **Ideal Low-Pass Filter:** Passes frequencies below a cutoff B ($|H(f)| = 1$) and blocks frequencies above B ($|H(f)| = 0$). Its $h(t)$ is a sinc function.
  - **Ideal High-Pass Filter:** Blocks frequencies below B and passes those above.
  - **Ideal Band-Pass Filter:** Passes only a specific band of frequencies.

- **V. Discrete-Time LTI Systems**

  - The same concepts apply directly to discrete signals $x[n]$.
  - **Time Domain:** The output is a **discrete convolution**:

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$

    . $h[n]$ is the response to a Kronecker delta $\delta[n]$.
  - **Frequency Domain (DFT):** The output spectrum is the product of the input spectrum and the system's frequency response: $Y[k] = H[k]X[k]$.
  - **Important Note:** For finite-length signals using the DFT, this multiplication corresponds to **circular convolution** in the time domain, not linear convolution.

Sampling Theorem

- **I. Motivation: From Continuous to Discrete**

  - Physical signals are often continuous-time (analog).
  - To process them on a computer, they must be converted to discrete-time signals.
  - This is done via **sampling**: observing the signal $x(t)$ at discrete, uniformly spaced time instants nT, creating the sequence $x[n] = x(nT)$.
  - **The Central Question:** Is it possible to perfectly recover the original analog signal $x(t)$ from its samples $x[n]$?

- **II. The Nyquist-Shannon Sampling Theorem**

  - **Statement:** If a signal $x(t)$ is **band-limited** to B Hz (contains no frequencies higher than B), it can be *completely determined* by its samples if the sampling rate F_s is at least $2B$ samples per second.
  - **Nyquist Rate:** The minimum required sampling rate, $2B$, is called the Nyquist rate.
  - **Nyquist Interval:** The maximum time between samples, $T = 1/(2B)$.

- **III. Frequency Domain Perspective: The Effect of Sampling**

  - **Key Property:** Sampling a signal in the time domain causes its spectrum to become **periodic** in the frequency domain.
  - The spectrum of the sampled signal, $X_s(f)$, consists of replicas of the original spectrum $X(f)$ repeated at integer multiples of the sampling frequency F_s.
  - **Three Scenarios:**
    1. $F_s \geq 2B$ **(Correct Sampling):** The spectral replicas **do not overlap**. The original spectrum can be perfectly recovered by applying an **ideal low-pass filter** to isolate the central replica.
    2. $F_s < 2B$ **(Undersampling):** The spectral replicas **overlap**. This phenomenon is called **aliasing**. High frequencies from the original signal "disguise" themselves as lower frequencies, and information is irretrievably lost. Perfect reconstruction is **impossible**.
    3. **Non-Band-limited Signal:** The spectrum extends infinitely, so aliasing is **inevitable** regardless of the sampling rate. In practice, an **anti-aliasing filter** (a low-pass filter) is applied to the analog signal *before* sampling to forcibly band-limit it.

- **IV. Time Domain Perspective: Signal Reconstruction**

  - **Interpolation:** The process of reconstructing the continuous signal from its samples.
  - **Reconstruction Formula:** The theorem provides the formula for perfect reconstruction, which is a **sinc interpolation**:

$$x(t) = \sum_{n=-\infty}^{\infty} x\left(\frac{n}{2B}\right) \operatorname{sinc}\left(2\pi B\left(t - \frac{n}{2B}\right)\right)$$

  - **Interpretation:** The original signal is reconstructed by summing an infinite series of sinc functions. Each sinc function is scaled by a sample value $x[n]$ and centered at the sample's time location nT. The sinc function acts as the **ideal interpolation kernel**, which corresponds to the impulse response of an ideal low-pass filter.

- **V. Conclusion**

  - The Sampling Theorem is a cornerstone of digital signal processing. It provides the theoretical justification for analog-to-digital conversion, defining the conditions under which the conversion can be performed without loss of information.
