4. Signal Transforms

In this chapter, we are going to analyze **signal transforms** other than the 1D Discrete Fourier Transform to describe (audiovisual) signals:

- The **2D Discrete Fourier Transform** (2D-DFT), as basi frequency representation for images.
- The **Discrete Cosine Transform** (DCT) and the **Karhunen-Loeve Transform** (KLT) as additional, specific representations for (audiovisual) signals.
- ~~The **Short-Time Fourier Transform** (STFT), as an extension of the DFT for the case of non-stationary signals.~~

4.1. 2D Discrete Fourier Transform

Introduction

Images can be modeled as **linear combinations** of simpler 2-variable functions; typically, **impulse functions**, as in the canonical representation, **complex exponentials** as in the Fourier representation, etc. The Fourier representation (and its underlying space/frequency model) is a useful tool to define a natural set of operations such as **Linear Space-Invariant (LSI)** operators. LSI operators linearly combine the pixel values in a given neighborhood of the pixel being processed (using **impulse responses** or **convolution masks**). They can be defined in the original (space) domain, through a **convolution**, and in the **transformed (frequency) domain**, through a **product**.

Space/Frequency Image processing tools

If LSI operators involve large impulse responses, they can be **efficiently implemented** thanks to the use of fast transforms:

1. The input image is fast transformed with the FFT.
2. The operation is performed in the transformed domain by a product.
3. The output image is obtained through an inverse FFT.

Space/Frequency image processing tools allow (among other things) **convolution operations**, **linear filter design**, **analysis of sampling** and **multi-resolution analysis**. To do all of this, we need to define a 2D Discrete Fourier Transform. But, how will we use it?

The Discrete Fourier Transform is going to be used to analyze the signals in the frequency domain. We cannot use the Fourier Transform, as it provides a **continuous function**, not suitable to work with a computer. So, we will use the DFT, as it provides a **sampled representation** of the FT in the frequency domain. Also, it is very fast, as it can be implemented with the FFT. In spite of all of this, how should we handle signals that may present a non-limited amount of samples?

- Modeling the actual (N samples) signal through **windowing** and accounting for the window transform.
- Using a time-dependent DFT, the **Short-Time Fourier Transform**.
- Processing the signal by blocks.

Definition of the 2D Discrete Fourier Transform

The 1D Discrete Fourier Transform is defined as:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N} n}, \quad 0 \leq k < N.$$

Recall that the DFT has symmetries and implicit periodicity. Its **extension to the 2D case** is defined as:

$$X[k,l] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] e^{-j2\pi \left(\frac{k}{M} m + \frac{l}{N} n \right)}, \quad 0 \leq k < M,\ 0 \leq l < N,$$

and therefore, its inverse is defined as:

$$x[m,n] = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k,l] e^{j2\pi \left(\frac{k}{M} m + \frac{l}{N} n \right)}, \quad 0 \leq m < M,\ 0 \leq n < N.$$

So, the 2D-DFT **transforms a 2D signal of $M \times N$ samples** in the original (space) domain into a set of $M \times N$ samples in the transformed (frequency) domain. Do not mistake it for the FT of a sequence $X(F_1, F_2)$

$$X(F_1, F_2) = \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} x[m,n] e^{-j2\pi(F_1 m + F_2 n)}.$$

$X(F_1, F_2)$ is a **periodic signal** in F_1 and F_2, with period $P = 1$. In the **DFT case**:

- Only data within a window is computed: $0 \leq m < M,\ 0 \leq n < N.$
- The transformed signal is sampled: $F_1 = \frac{k}{M}$ and $F_2 = \frac{l}{N}$.

So, **the DFT is a sampled version of the FT computed over a windowed signal**. Usually, samples in the spatial domain are real (or integer) whereas samples in the frequency domain are complex. The DFT of a signal can be represented in terms of its **magnitude** (modulus) and **phase**,

$$|X[k,l]| = \sqrt{\Re^2 X[k,l] + \Im^2 X[k,l]}, \qquad \varphi_X[k,l] = \arctan \frac{\Im X[k,l]}{\Re X[k,l]}.$$

The original samples **can always be recovered** from the transformed ones:

$$DFT^{-1}\left\{ DFT\{x[m,n]\} \right\} = x[m,n], \quad 0 \leq m < M,\ 0 \leq n < N.$$

Usually, only the magnitude is represented, although the phase carries more information. To represent the magnitude, a **logarithmic transform** is commonly applied, due to its dynamic range.

Basic properties

Some signal transforms