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 a basic 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:

$$X[k]=\sum_{n=0}^{N-1} x[n] e^{-j 2 \pi \frac{k}{N} n}, \quad 0 \leq k<N .$$

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

$$X[k, l]=\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] e^{-j 2 \pi\left(\frac{k}{M} m+\frac{l}{N} n\right)}, \quad 0 \leq k<M,\ 0 \leq l<N,$$

and therefore, its inverse is defined as:

$$x[m, n]=\frac{1}{M N} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] e^{j 2 \pi\left(\frac{k}{M} m+\frac{l}{N} n\right)}, \quad 0 \leq m<M,\ 0 \leq n<N .$$

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 $X(F_1, F_2)$

$$X(F_1, F_2)=\sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} x[m, n] e^{-j 2 \pi(F_1 m+F_2 n)} .$$

$X(F_1, F_2)$ 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:

$$D F T^{-1}\{D F T\{x[m, n]\}\}=x[m, n], \quad 0 \leq m<M,\ 0 \leq n<N .$$

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

DFT properties

With respect to the Fourier Transform, the DFT has some similar properties. But, in the case of a DFT of length L (N in 1D and (M, N) in 2D), indexes should remain in the interval $[0, L-1]$. This leads to circular convolution, displacement or time-reversal representations. Thus, properties can be expressed with the help of

$$\tilde{t}[m, n]=\sum_{r \in \mathbb{Z}} \sum_{s \in \mathbb{Z}} t[m+r M, n+s N], \quad \begin{cases}0 \leq m \leq M-1 \\ 0 \leq n \leq N-1\end{cases}$$

- **Convolution:** spatial convolution implies frequency product:

$$t[m, n]=x[m, n] * y[m, n] \Longrightarrow \tilde{t}[m, n] \leftrightarrow X[k, l] \cdot Y[k, l]$$

- **Convolution and windowing:** windowing in the spatial domain implies frequency convolution:

$$T[k,l] = X[k,l] * Y[k,l] \implies x[m,n] \cdot y[m,n] \leftrightarrow \frac{1}{MN}\tilde{T}[k,l]$$

- **Spatial shift or translation:** a spatial shift only affects the phase of the transformed signal:

$$\tilde{x}[m-m', n-n'] \leftrightarrow X[k,l]e^{-j2\pi\left(\frac{k}{M}m' + \frac{l}{N}n'\right)}$$

- **Frequency shift or modulation:** multiplication of an image by a complex exponential implies a frequency shift:

$$x[m,n]e^{j2\pi\left(\frac{k'}{M}m + \frac{l'}{N}n\right)} \leftrightarrow \tilde{X}[k-k', l-l']$$

- **Separability:** since the 2D-DFT kernel is separable, the 2D-DFT can be implemented as two 1D-DFT:

$$\begin{aligned} X[k,l] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]e^{-j2\pi\left(\frac{k}{M}m + \frac{l}{N}n\right)} = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]e^{-j2\pi\frac{k}{M}m}e^{-j2\pi\frac{l}{N}n} = \\ \sum_{m=0}^{M-1} e^{-j2\pi\frac{k}{M}m} \sum_{n=0}^{N-1} x[m,n]e^{-j2\pi\frac{l}{N}n} \\ \implies DFT^{2D}_{\text{image}}[\cdot] = DFT^{1D}_{\text{rows}}\left[DFT^{1D}_{\text{columns}}[\cdot]\right] = DFT^{1D}_{\text{columns}}\left[DFT^{1D}_{\text{rows}}[\cdot]\right]. \end{aligned}$$

- **Rotation:** a spatial rotation corresponds to the same frequency rotation:

$$\left.\begin{array}{ll} m = r\cos\theta & n = r\sin\theta \\ k = \rho\cos\varphi & l = \rho\sin\varphi \end{array}\right\} \implies x[r, \theta + \theta_0] \leftrightarrow X[\rho, \varphi + \theta_0]$$

- **Hermitian symmetry:** if the image is **real**, its DFT presents Hermitian symmetry:

$$X[k,l] = X^*[-k,-l].$$

Note the implicit periodicity of the DFT. This is very useful for filter implementation.

In the $M \times N$ support, the symmetries are:

$$\begin{aligned} &\to X[0,0] = X^*[0,0] \\ &\to X[0,l] = X^*[0, N-l], \qquad 1 \le l < N \\ &\to X[k,0] = X^*[M-k, 0], \qquad 1 \le k < M \\ &\to X[k,l] = X^*[M-k, N-l], \qquad \begin{cases} 1 \le k < M, \\ 1 \le l < N \end{cases} \end{aligned}$$

Other 2D-DFT properties

Since images are generally positive signals, the highest value of the magnitude of their transform is at $(k,l) = (0,0)$. Given the DFT symmetries, the four corners of the transformed image contain the highest values of the magnitude. In order to **help visualization**, the transformed image is represented centered at $\left(\frac{M}{2}, \frac{N}{2}\right)$. This can be seen as an example of the modulation property:

$$x[m,n]e^{j2\pi\left(\frac{k'}{M}m + \frac{l'}{N}n\right)} \leftrightarrow \tilde{X}[k-k', l-l']$$

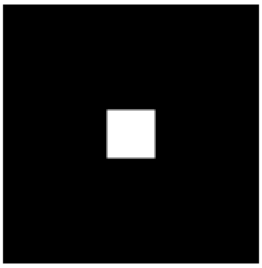
For $k' = \frac{M}{2}, l' = \frac{N}{2}$, we have that

$$x[m,n]e^{j2\pi\left(\frac{m}{2} + \frac{n}{2}n\right)} \leftrightarrow \tilde{X}\left[k - \frac{M}{2}, l - \frac{N}{2}\right],$$

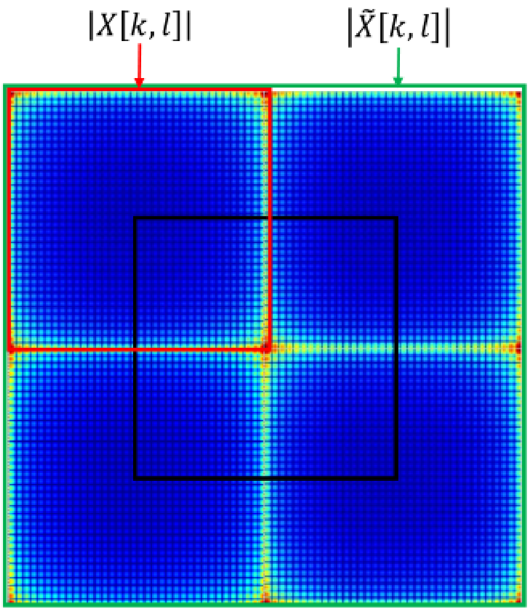
so

$$x[m,n]e^{j\pi(m+n)} = x[m,n](-1)^{m+n} \leftrightarrow \tilde{X}\left[k - \frac{M}{2}, l - \frac{N}{2}\right].$$

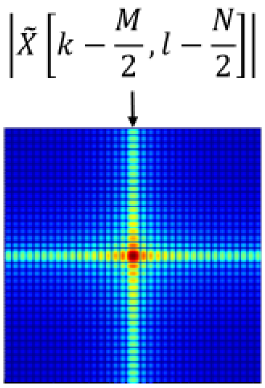
For an example, imagine that we have the following image:



Its 2D-DFT (in magnitude is):



So, we modulate the transformation and we take only the **center square** of the transformed image:



Some signal transforms

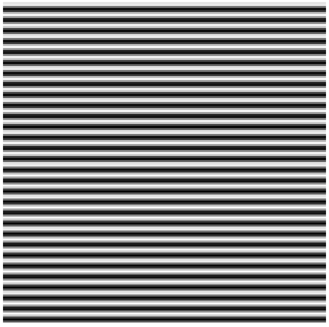
Basic signal transforms

If a 2D function is **separable** into a product of 1D functions, so is its DFT:

$$X[k,l] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]e^{-j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)} = \Big[x[m,n] = x_1[m]x_2[n]\Big] =$$

$$\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x_1[m]x_2[n]e^{-j2\pi\left(\frac{k}{M}m+\frac{l}{N}n\right)} = \sum_{m=0}^{M-1} x_1[m]e^{-j2\pi\frac{k}{M}m}\sum_{n=0}^{N-1} x_2[n]e^{-j2\pi\frac{l}{N}n} = \boxed{X_1[k]X_2[l].}$$

Example: Imagine that we have an image like this:



with $M = N = 300, L = 30$. The image can be represented as:

$$x[m,n] = \sin(2\pi F_1 m + 2\pi F_2 n),\ F_1 = 0, F_2 = 0.1 \implies$$
$$x[m,n] = \sin(2\pi F_2 n) = 1 \cdot \sin(2\pi F_2 n) = x_1[m] \cdot x_2[n].$$

Then, as it is a product of 1D functions, its DFT is

$$x[m,n] = x_1[m]x_2[n] \leftrightarrow X[k,l] = X_1[k]X_2[l].$$

So, let's calculate their DFTs separately: we have, on the one hand,

$$x_1[m] = 1 \implies X_1[k] = \sum_{m=0}^{M-1} 1 \cdot e^{-j2\pi\frac{k}{M}m} = \left[\sum_{n=0}^{P} a^n = \frac{1-a^{P+1}}{1-a}, \forall a \neq 1\right] =$$
$$= \left\{\begin{array}{c} k = 0 \implies \sum_{m=0}^{M-1} 1 = M \\ k \neq 0 \implies \frac{1-e^{-j2\pi k\frac{M}{M}}}{1-e^{-j2\pi\frac{k}{M}}} = 0 \end{array}\right\} = \boxed{M \cdot \delta[k].}$$

and on the other hand,

$$x_2[n] = \sin(2\pi F_2 n) = \left[F_2 = \frac{L_2}{N}\right] = \frac{e^{j2\pi\frac{L_2 n}{N}} - e^{-j2\pi\frac{L_2 n}{N}}}{2j}.$$

We know that, because of the modulation property,

$$1 \leftrightarrow N\delta[l], \qquad e^{j2\pi\frac{L_2 n}{N}} \leftrightarrow N\delta[l-L_2].$$

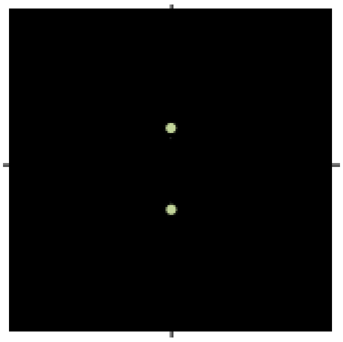
Then, the DFT of this signal is

$$\boxed{X_2[l] = \frac{N}{2j}\Big(\delta[l-L_2] - \delta[l+L_2]\Big).}$$

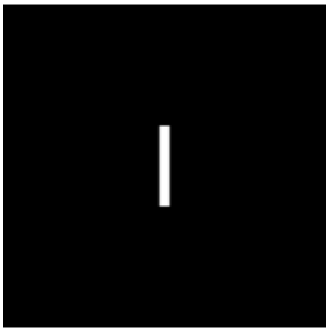
Hence, the complete DFT is

$$\boxed{X[k,l] = X_1[k]X_2[l] = \frac{MN}{2j}\delta[k]\Big(\delta[l-L_2] - \delta[l+L_2]\Big).}$$

Its representation yields



Example: we have the following image:



which is represented via

$$x[m,n] = \Pi_{M'}[m-m_0]\Pi_{N'}[n-n_0], \quad N' > M'.$$

So, a product of pulses of length M' and N' centered at m_0 and n_0 respectively. As a shift only affects the phase, we will calculate its separate DFTs without paying much attention to these m_0 and n_0:

$$X_1[k] = \sum_{m=0}^{M_1} \Pi_{M'}[m] e^{-j2\pi \frac{k}{M} m} = \sum_{m=0}^{M'-1} e^{-j2\pi \frac{k}{M} m} = \frac{1 - e^{-j2\pi \frac{k}{M} M'}}{1 - e^{-j2\pi \frac{k}{M}}} =$$
$$\frac{e^{-j\pi \frac{k}{M} M'}}{e^{-j\pi \frac{k}{M}}} \cdot \left[\frac{e^{j\pi \frac{k}{M} M'} - e^{-j\pi \frac{k}{M} M'}}{e^{j\pi \frac{k}{M}} - e^{-j\pi \frac{k}{M}}} \right] \implies$$
$$\implies |X_1[k]| = \left| \frac{\sin\left[\pi k \frac{M'}{M}\right]}{\sin\left[\pi \frac{k}{M}\right]} \right| \leftarrow \text{Discrete sinc}$$

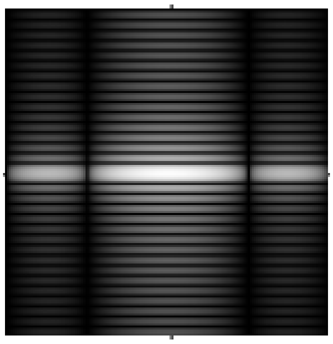
This signal is 0 at $k = \frac{M}{M'}$, and its maximum lies at 0, where its value is

$$X_1[0] \approx \frac{\pi k M'/M}{\pi k/M} = M'.$$

Applying the separability, the whole signal DFT is

$$|X[k,l]| = \left| \frac{\sin\left[\pi k \frac{M'}{M}\right]}{\sin\left[\pi \frac{k}{M}\right]} \right| \left| \frac{\sin\left[\pi l \frac{N'}{N}\right]}{\sin\left[\pi \frac{l}{N}\right]} \right|.$$

Its representation yields:



If we had $M' = 1, N' = N$, we would have a **vertical line segment**, and its transform would be an extreme case of the previous one,

$$
\begin{gathered}
|X_1[k]| = \left|\frac{\sin[\pi\frac{k}{M}]}{\sin[\pi\frac{k}{M}]}\right| = 1, \\
|X_2[l]| = \left|\frac{\sin[\pi l]}{\sin[\pi\frac{l}{N}]}\right| = N \cdot \delta[l], \\
|X[k,l]| = |X_1[k]| \cdot |X_2[l]| = 1 \cdot N \cdot \delta[l] = \boxed{N \cdot \delta[l].}
\end{gathered}
$$

This transform is a **horizontal line segment**. So, we can see that a given line in the spatial domain is related to a **perpendicular line** in the frequency domain.

4.2. Linear Filtering

Introduction

A Linear Space-Invariant operator is characterized by its impulse response $h[m,n]$, and the output of an operator of this kind can be computed using **convolution**:

$$y[m,n] = \sum_{m'}\sum_{n'} x[m',n']h[m-m',n-n'] = \sum_{m'}\sum_{n'} x[m-m',n-n']h[m',n'] = x[m,n] * h[m,n].$$

In the frequency domain, the output transform is defined by the **product** of the input transform and the **frequency response** $H[k,l]$ of the system,

$$y[m,n] = x[m,n] * h[m,n] \implies \tilde{y}[m,n] \leftrightarrow Y[k,l] = X[k,l] \cdot H[k,l]$$

For each pixel position (m,n),

- A **2D-mirrored version** of $h[m',n']$ is shifted at the pixel position (m,n): $h[m-m',n-n']$.
- The output $y[m,n]$ is the sum of the pixel values included in the neighborhood defined by the filter and weighted by the co-located filter values. A **similar interpretation** where the input signal is shifted is also possible.

The computation of the output valeus **close to the image border** requires the use of pixel values outside the input image support. These values are commonly introduced by:

- **Zero padding:** including a frame of zeros in the input signal. It may introduce discontinuities.
- **Mirroring:** including pixel values that are a symmetric mirroring of the input. It generates a smooth transition.
- **Translation variant filter:** to only use known pixel values. So, the filter is not invariant in the border of the image, but behaves in a similar way using only the values it can access.

The result of the convolution of an image of size $M \times N$ and a filter with impulse response of size $H_1 \times H_2$ is an image of size $(M + H_1 - 1) \times (N + H_2 - 1)$. Nevertheless, in almost all practical cases, the output image support is **finally restricted** to the original image size.

In some cases, the 2D impulse response can be defined as the product of two 1D impulse responses (**separability**): this allows for faster implementations, as

$$y[m,n] = \sum_{m'}\sum_{n'} x[m',n']h[m-m',n-n'] = \sum_{m'}\sum_{n'} x[m',n']h_1[m-m']h_2[n-n'] = \\ \sum_{n'} h_2[n-n'] \sum_{m'} x[m',n']h_1[m-m'] = x[m,n] *^{1D}_{\text{rows}} h_1[m] *^{1D}_{\text{columns}} h_2[n].$$

The frequency response of a 2D separable filter can be defined as the **product of the two 1D frequency responses**, $H[k,l] = H_1[k] \cdot H_2[l]$.

Filters are defined in the spatial domain when their **size** (area of support) is relatively **small**. In other cases, filters are usually defined in the frequency domain. They are commonly **square, odd in size and symmetric**. We are going to study two different types of filters:

- **Low-pass filters**, that perform a spatial average, and their application to noise removal.
- **High-pass filters**, that perform an estimation of the derivative, and their application to contour detection.

Low-pass filters designed in the spatial domain

We are going to study low-pass filters in the context of **noise reduction**. In image processing, there are **three main types of noise**:

- **Gaussian noise:** it can appear in the sensors of a CCD camera.

$$p(n) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(n-\mu)^2}{\sigma^2}}$$

- **Uniform noise:** it is produced by the quantization process.

$$p(n) = \begin{cases} \frac{1}{b-a}, & \text{if } a \le n \le b \\ 0, & \text{otherwise} \end{cases}$$

- **Salt and pepper noise:** typical of error transmissions or damaged CCD sensors.

$$\tilde{i}(m,n) = \begin{cases} s, & \text{with probability } p \\ i(m,n), & \text{with probability } 1-p \end{cases}$$

s is a random variable with values in $\{0, 255\}$.