2D Transforms
- Introduction to 2D signal processing
- 2D Fourier Representations

2D Fourier Representations
From "Continuous Time" to "Continuous Space."

One dimensional CTFT:
\[ F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi \omega t} \, dt \]
\[ f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j2\pi \omega t} \, d\omega \]

Two dimensional CTFT:
\[ F(\omega_x, \omega_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j(\omega_x x + \omega_y y)} \, dx \, dy \]
\[ f(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\omega_x, \omega_y) e^{j(\omega_x x + \omega_y y)} \, d\omega_x \, d\omega_y \]
integrals \rightarrow double integrals; sum of x and y exponents in kernal function.

One dimensional DFT:
\[ F[k] = \frac{1}{N} \sum_{n=0}^{N-1} f[n] e^{-j2\pi \frac{k}{N} n} \]
\[ f[n] = \sum_{k=0}^{N-1} F[k] e^{j2\pi \frac{k}{N} n} \]

Two dimensional DFT:
\[ F[k_x, k_y] = \frac{1}{N_x N_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} f[n_x, n_y] e^{-j\left(\frac{2\pi k_x}{N_x} n_x + \frac{2\pi k_y}{N_y} n_y\right)} \]
\[ f[n_x, n_y] = \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} F[k_x, k_y] e^{j\left(\frac{2\pi k_x}{N_x} n_x + \frac{2\pi k_y}{N_y} n_y\right)} \]
double sums; sum of x and y exponents in kernal function.

Importance of Orthogonality
Fourier series represent periodic signals as weighted sum of basis functions.
\[ f[n] = \sum_{k=0}^{N-1} F[k] e^{j \frac{2\pi k}{N} n} \]
We “sifted” out the \( f^k \) component by multiplying both sides by \( e^{-j \frac{2\pi k}{N} n} \) and summing over a period.
\[ N \sum_{n=0}^{N-1} f[n] e^{-j \frac{2\pi k}{N} n} = \sum_{n=0}^{N-1} F[k] e^{j \frac{2\pi k}{N} n} e^{-j \frac{2\pi k}{N} n} = \sum_{k=0}^{N-1} F[k] \sum_{n=0}^{N-1} e^{j \frac{2\pi k}{N} (n-k)} \]
\[ = \begin{cases} NF[k] & \text{if } k = l \\ 0 & \text{otherwise} \end{cases} \]
This sifting provided an explicit “analysis” formula for the coefficients:
\[ F[k] = \frac{1}{N} \sum_{n=0}^{N-1} f[n] e^{-j \frac{2\pi k}{N} n} \]
Orthogonality of the basis functions is key to Fourier decomposition.
Orthogonality
The form of the 2D Fourier kernel preserves orthogonality.

1D DFT basis functions: \( \phi_l[n] = e^{\frac{2i\pi ln}{N}} \)

“Inner product” of 1D basis functions:
\[
\sum_n \phi^*_l[n] \phi[m] = \sum_n e^{-\frac{2i\pi ln}{N} - \frac{2i\pi lm}{N}} = N \delta[l - m]
\]

2D DFT basis functions: \( \phi_{kx,ky}[n_x, n_y] = e^{\frac{2i\pi (kx n_x + ky n_y)}{NxNy}} \)

“Inner product” of 2D basis functions:
\[
\sum_{n_x,n_y} \phi_{kx,ky}^*[n_x, n_y] \phi_{kx',ky'}[n_x, n_y] = \sum_{n_x,n_y} e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy} - \frac{2i\pi (kx' n_x + ky' n_y)}{NxNy}} = NxNy \delta[kx - kx'] \delta[ky - ky']
\]

Check Yourself
The 2D Fourier basis functions have the following form.
\( \phi_{kx,ky}[n_x, n_y] = e^{\frac{2i\pi (kx n_x + ky n_y)}{NxNy}} \)

Which (if any) of the following images show the real part of one of the basis functions \( \phi_{kx,ky}[n_x, n_y] \)?

A B C D

What values of \( k_x \) and \( k_y \) correspond to basis function?

2D Discrete Fourier Transform
Example: Find the DFT of a 2D unit sample.

\( f_0[n_x, n_y] = \delta[n_x] \delta[n_y] = \begin{cases} 1 & n_x = 0 \text{ and } n_y = 0 \\ 0 & \text{otherwise} \end{cases} \)

\[
F_0[kx, ky] = \frac{1}{N_xN_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} \delta[n_x] \delta[n_y] e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy}} = \frac{1}{N_xN_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy}} = \frac{1}{N_xN_y} \delta[kx] \delta[ky]
\]

2D Discrete Fourier Transform
Alternatively, implement a 2D DFT as a sequence of 1D DFTs.

\[
F[kx, ky] = \frac{1}{N_xN_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} f[n_x, n_y] e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy}}
\]

\[
= \frac{1}{N_y} \sum_{n_y=0}^{N_y-1} \left( \frac{1}{N_x} \sum_{n_x=0}^{N_x-1} f[n_x, n_y] e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy}} \right) e^{-\frac{2i\pi ky n_y}{Ny}}
\]

First take DFTs of rows

Then take DFTs of resulting columns

Start with a 2D function of time \( f[n_x, n_y] \).
- Replace each row by the DFT of that row.
- Replace each column by the DFT of that column.
The result is \( F[kx, ky] \), the 2D DFT of \( f[n_x, n_y] \).

Could just as well start with columns and then do rows.

2D Discrete Fourier Transform
Example: Find the DFT of a constant.

\( f_1[n_x, n_y] = 1 \)

\[
F_1[kx, ky] = \frac{1}{N_xN_y} \sum_{n_x=0}^{N_x-1} \sum_{n_y=0}^{N_y-1} 1 e^{-\frac{2i\pi (kx n_x + ky n_y)}{NxNy}}
\]

\[
= \left( \frac{1}{N_x} \sum_{n_x=0}^{N_x-1} e^{-\frac{2i\pi kx n_x}{Nx}} \right) \left( \frac{1}{N_y} \sum_{n_y=0}^{N_y-1} e^{-\frac{2i\pi ky n_y}{Ny}} \right) = \delta[kx] \delta[ky]
\]

1 \( \iff \) \( \delta[kx] \delta[ky] \)
Effect of image translation (shifting) on its Fourier transform.

Assume that \( f_0[n_x, n_y] = \delta[n_x] \) and \( F_0[k_x, k_y] \).

Find the 2D DFT of \( f_1[n_x, n_y] = f_0[n_x-n_x_0, n_y-n_y_0] \):

\[
F_1[k_x, k_y] = \sum_{n_x} \sum_{n_y} f_1[n_x, n_y] e^{-j2\pi k_x n_x} e^{-j2\pi k_y n_y}
\]

\[
= \sum_{k_x} \sum_{k_y} f_0[n_x-n_x_0, n_y-n_y_0] e^{-j2\pi k_x n_x} e^{-j2\pi k_y n_y}
\]

Let \( l_x = n_x-n_x_0 \) and \( l_y = n_y-n_y_0 \). Then

\[
F_1[k_x, k_y] = \sum_{l_x} \sum_{l_y} f_0[l_x, l_y] e^{-j2\pi k_x (l_x+l_x_0)} e^{-j2\pi k_y (l_y+l_y_0)}
\]

\[
e^{-j2\pi l_x k_x} e^{-j2\pi l_y k_y} F_0[k_x, k_y]
\]

Translating an image adds linear (in \( k_x, k_y \)) phase to its transform.
Example: Find the DFT of a shifted 2D unit sample.

Using Python
Calculating DFTs is most efficient in NumPy (Numerical Python).
- NumPy arrays are homogeneous: their elements are of the same type
- Numpy operators (+, -, abs, .real, .imag) combine elements to create new arrays. e.g., (f+g)[n] = f[n]+g[n].
- 2D Numpy arrays can be indexed by tuples: e.g., f[r,c] = f[r][c].
- 2D Numpy arrays support negative indices: e.g., f[-1] = f[len(f)-1]
- 2D indices address row then column.

Numpy Example
Make a white square on a black background.

Big and Small
Triangle

What are the dominant features of the magnitude of the DFT of a triangle?
Ocean
What are the dominant features of the DFT magnitude of an ocean view?

Trees
What are the dominant features of the DFT magnitude of these trees?

Moon
What are the dominant features of the DFT magnitude of the moon?

Bricks
What are the dominant features of the DFT magnitude of this brick wall?

Fingerprint
What are the dominant features of the DFT magnitude of this fingerprint?

Summary
Introduced 2D signal processing.
- generally simple extensions of 1D ideas

Introduced 2D Fourier representations.
- Fourier kernel comprises the sum of an $x$ part and a $y$ part
- basis functions are complex exponentials

Properties of 2D DFT
- transform all of the rows then transform all of the columns
- transform all of the columns then transform all of the rows